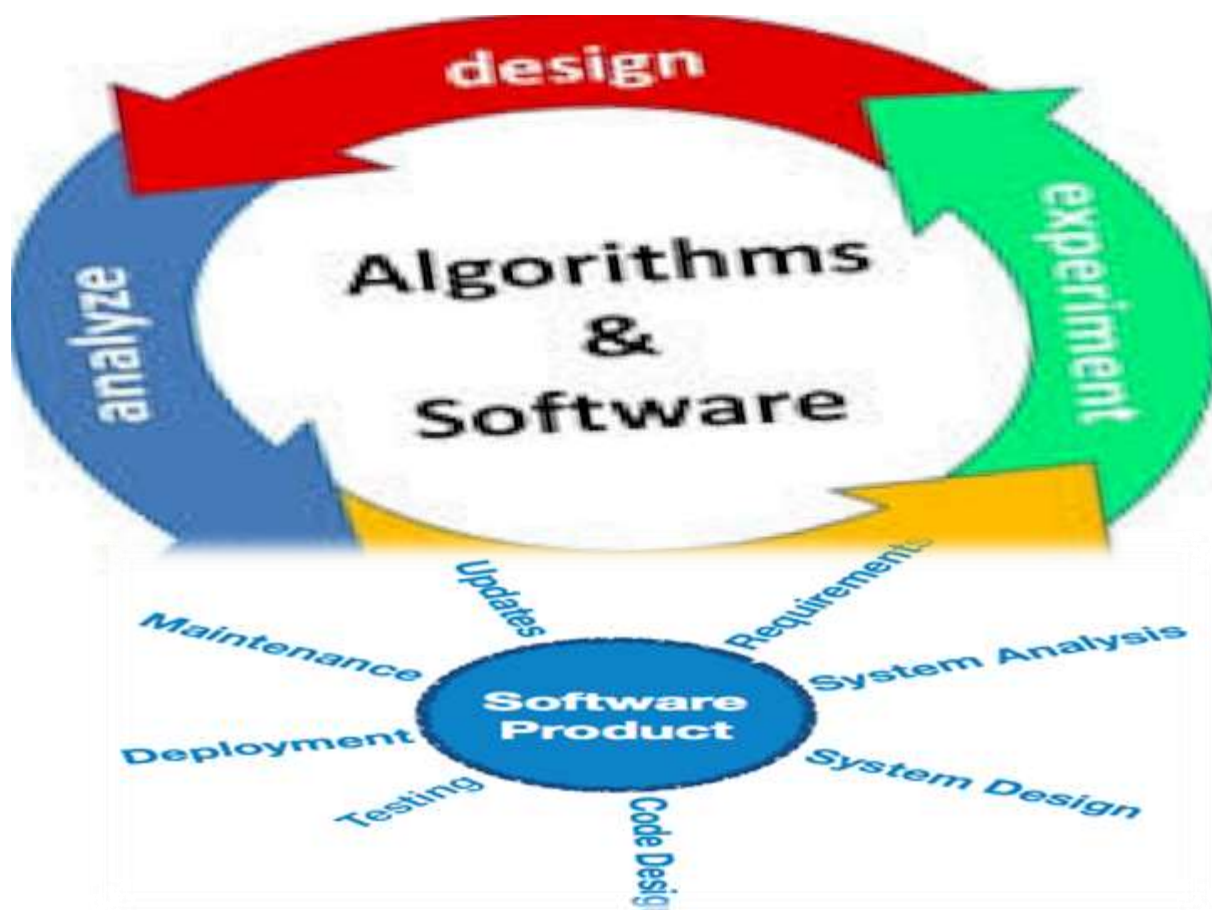


ЎЗБЕКИСТОН РЕСПУБЛИКАСИ
ОЛИЙ ВА ЎРТА МАХСУС ТАЪЛИМ ВАЗИРЛИГИ

МУҲАММАД АЛ-ХОРАЗМИЙ НОМИДАГИ ТОШКЕНТ АХБОРОТ
ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ ҲУЗУРИДАГИ ПЕДАГОГ КАДРЛАРНИ
ҚАЙТА ТАЙЁРЛАШ ВА УЛАРНИНГ МАЛАКАСИНИ ОШИРИШ
ТАРМОҚ МАРКАЗИ



ЎҚУВ – УСЛУБИЙ МАЖМУА

SQL СЎРОВЛАР ТИЛИДА МАЪЛУМОТЛАР БАЗАЛАРИНИ ЯРАТИШ

“Дастурий инжиниринг” йўналиши

**ЎЗБЕКИСТОН РЕСПУБЛИКАСИ
ОЛИЙ ВА ЎРТА МАХСУС ТАЪЛИМ ВАЗИРЛИГИ**

**ОЛИЙ ТАЪЛИМ ТИЗИМИ ПЕДАГОГ ВА РАЎБАР КАДРЛАРИНИ
ҚАЙТА ТАЙЁРЛАШ ВА УЛАРНИНГ МАЛАКАСИНИ ОШИРИШНИ
ТАШКИЛ ЭТИШ БОШ ИЛМИЙ - МЕТОДИК МАРКАЗИ**

**МУЎАММАД АЛ-ХОРАЗМИЙ НОМИДАГИ ТОШКЕНТ АХБОРОТ
ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ ҲУЗУРИДАГИ ПЕДАГОГ
КАДРЛАРНИ ҚАЙТА ТАЙЁРЛАШ ВА УЛАРНИНГ МАЛАКАСИНИ
ОШИРИШ ТАРМОҚ МАРКАЗИ**

“Дастурий инжиниринг” йўналиши

**“SQL СЎРОВЛАР ТИЛИДА МАЪЛУМОТЛАР
БАЗАЛАРИНИ ЯРАТИШ”**

МОДУЛИ БЎЙИЧА

Ў Қ У В – У С Л У Б И Й М А Ж М У А

Тошкент - 2022

**Мазкур ўқув-услубий мажмуа Олий ва ўрта махсус таълим
вазирлигининг 2020 йил 7 декабрдаги 648-сонли буйруғи билан
тасдиқланган ўқув режа ва дастур асосида тайёрланди.**

Тузувчилар: ТАТУ, Информатика
кафедраси мудири, проф.

Б.Мўминов

Такризчилар: Беларусь-Ўзбекистон кўшма тармоқлараро амалий техник
квалификациялар институти, илмий ишлар ва инновациялар
бўйича директор ўринбосари в.б., доц. Л.Набиулина,
Муҳаммад ал-Хоразмий номидаги ТАТУ, “Ахборот
технологиялари” кафедраси мудири, проф. Х.Зайнидинов.

**Ўқув -услубий мажмуа Тошкент ахборот технологиялари
университети Кенгашининг 2020 йил 29 октябрдаги 3(705)- сонли қарори
билан маъқулланган.**

МУНДАРИЖА

I. Ишчи дастур	4
II. Фойдаланиладиган интерфаол таълим методлари.....	11
III. Назарий материаллар.....	18
IV. Амалий машғулот материаллари.....	110
V. Кейслар банки.....	147
VI. Глоссарий	150
VII. Адабиётлар рўйхати.....	156

І БЇЛИМ

ИШЧИ ДАСТУР

I. ИШЧИ ДАСТУР

Кириш

Дастур Ўзбекистон Республикасининг 2020 йил 23 сентябрда тасдиқланган “Таълим тўғрисида”ги Қонуни, Ўзбекистон Республикаси Президентининг 2017 йил 7 февралдаги “Ўзбекистон Республикасини янада ривожлантириш бўйича Ҳаракатлар стратегияси тўғрисида”ги ПФ-4947-сон, 2019 йил 27 августдаги “Олий таълим муассасалари раҳбар ва педагог кадрларининг узлуксиз малакасини ошириш тизимини жорий этиш тўғрисида”ги ПФ-5789-сон, 2019 йил 8 октябрдаги “Ўзбекистон Республикаси олий таълим тизимини 2030 йилгача ривожлантириш концепциясини тасдиқлаш тўғрисида”ги ПФ-5847-сон ва 2020 йил 29 октябрдаги “Илм-фанни 2030 йилгача ривожлантириш концепциясини тасдиқлаш тўғрисида”ги ПФ-6097-сонли Фармонлари ҳамда Ўзбекистон Республикаси Вазирлар Маҳкамасининг 2019 йил 23 сентябрдаги “Олий таълим муассасалари раҳбар ва педагог кадрларининг малакасини ошириш тизимини янада такомиллаштириш бўйича қўшимча чора-тадбирлар тўғрисида”ги 797-сонли Қарорида белгиланган устувор вазифалар мазмунидан келиб чиққан ҳолда тузилган бўлиб, у олий таълим муассасалари педагог кадрларининг касб маҳорати ҳамда инновацион компетентлигини ривожлантириш, соҳага оид илғор хорижий тажрибалар, янги билим ва малакаларни ўзлаштириш, шунингдек амалиётга жорий этиш кўникмаларини такомиллаштиришни мақсад қилади.

Қайта тайёрлаш ва малака ошириш йўналишининг ўзига хос хусусиятлари ҳамда долзарб масалаларидан келиб чиққан ҳолда дастурда тингловчиларнинг мутахассислик фанлар доирасидаги билим, кўникма, малака ҳамда компетенцияларига қўйиладиган талаблар такомиллаштирилиши мумкин.

Модулнинг мақсади ва вазифалари

“SQL сўровлар тилида маълумотлар базаларини яратиш” модулининг мақсад ва вазифалари:

- SQL сўровлар тилида маълумотлар базаларини яратиш технологиясини ўрганиш;
- SQL сўровлар тилида маълумотлар базаларини яратиш босқичлар ва уларни амалиётга қўллаш касбий кўникмаларини шакллантириш;

Модул бўйича тингловчиларнинг билими, кўникмаси, малакаси ва компетенцияларига қўйиладиган талаблар

“SQL сўровлар тилида маълумотлар базаларини яратиш” модулини ўзлаштириш жараёнида амалга ошириладиган масалалар доирасида:

Тингловчи:

- маълумотлар ва уларнинг турлари, маълумотлар базаси архитектураси, SQL сўровлар тили операторлари, маълумотлар базасини лойиҳалаштириш босқичлари, сўровлар ва уларни қайта ишлаш босқичлари, транзакция хусусиятлари, концепцияси ва модели, маълумотларни йиғиш, жамлаш ва излаш усулларини билиши керак.

- SQL тилида маълумотлар базасини яратиш, ER диаграммасини тузиш, сўровларни амалга ошириш ва транзакцияларни яратиш *кўникмаларига* эга бўлиши лозим.

- маълумотлар базасини яратиш ва уларни бошқариш тизимлари билан ишлаш *малакаларига* эга бўлиши лозим.

- мақсадга йўналтирилган маълумотлар базаларини лойиҳалаш, маълумотлар базасини яратиш ва бошқариш *компетенцияларига* эга бўлиши лозим.

Модулни ташкил этиш ва ўтказиш бўйича тавсиялар

“SQL сўровлар тилида маълумотлар базаларини яратиш” модули маъруза ва амалий машғулотлар шаклида олиб борилади.

Курсни ўқитиш жараёнида таълимнинг замонавий методлари, педагогик технологиялар ва ахборот-коммуникация технологиялари қўлланилиши назарда тутилган:

- маъруза дарсларида замонавий компьютер технологиялари ёрдамида презентацион ва электрон-дидактик технологиялардан;

- ўтказиладиган амалий машғулотларда техник воситалардан, экспресс-сўровлар, тест сўровлари, ақлий ҳужум, гуруҳли фикрлаш, кичик гуруҳлар билан ишлаш ва бошқа интерактив таълим усулларини қўллаш назарда тутилади.

Модулнинг ўқув режадаги бошқа модуллар билан боғлиқлиги ва узвийлиги

“SQL сўровлар тилида маълумотлар базаларини яратиш” модули мазмуни ўқув режадаги “Катта маълумотларни қайта ишлаш усул ва воситалар”, “Операцион тизимлар”, “Булутли ҳисоблаш технологиялари” модуллари билан узвий боғланган ҳолда педагогларнинг ўрнатилган тизимларда турли схемаларни амалга ошириш ва иловалар яратиш бўйича касбий педагогик тайёргарлик даражасини оширишга хизмат қилади.

Модулнинг олий таълимдаги ўрни

Модулни ўзлаштириш орқали тингловчилар бошқарув масалаларини ечишда ўрнатилган тизилардан амалда тўғри фойдаланиш, қўллаш ва баҳолашга доир касбий компетентликка эга бўладилар.

Модул бўйича соатлар тақсимоги

№	Модуль мавзулари	Аудитория укув юкламаси			
		Жами	жумладан		
			Назарий	Амаий машғуло	Кучма машғуло
1.	Маълумотлар ва уларнинг турлари. Маълумотлар базаси архитектураси	2	2		
2.	SQL сўровлар тили	2	2		
3	Маълумотлар базасини лойиҳалаштириш.	2	2		
4	Сўровлар ва уларни қайта ишлаш босқичлари.	2	2		
5	Транзакция. Транзакция концепцияси ва модели.	4	2		
6	Маълумотларни йиғиш, жамлаш ва излаш усуллари.	2	2		
7	SQL тилида маълумотлар базасини яратиш.	4		4	
8	SQL тилида ER диаграммасини яратиш.	2		2	
9	SQL тили асосида сўровларни амалга ошириш.	2		2	
10	SQL тили асосида транзакцияни яратиш.	2		2	
11	SQL тили асосида маълумотларни йиғиш, жамлаш ва излаш усуллари	2		2	
	Жами:	22	12	12	0

НАЗАРИЙ МАШҒУЛОТЛАР МАЗМУНИ

1 - мавзу: Маълумотлар ва уларнинг турлари. Маълумотлар базаси архитектураси (2 соат)

SQL тили имкониятлари. Маълумотлар базасини бошқариш тизими. Маълумотларнинг кўриниши. Маълумот моделлари. Маълумотлар базаси архитектураси. Транзакцияларни бошқариш. Реляцион модел. Реляцион маълумотлар базаси структураси. Маълумотлар базаси схемаси. Модел схемаси. Реляцион сўровлар тили. Реляционоперациялар. Калитлар. Реляцион алгебра.

2 - мавзу: SQL сўровлар тили (2 соат)

Кўп сонли жадваллар устида сўровлар билан ишлашни амалга ошириш принциплари. Маълумотларни саралаш усуллари. Бирлаштириш функциялари. SQL маълумотларни таърифлари. SQL да union, intersect va exсert амаллари билан танишиш Union амали хоссалари.

3 – мавзу: Маълумотлар базасини лойиҳалаштириш (2 соат)

Малумотлар базасини лойиҳалаш усуллари. Лойиҳалаш жараёни муаммолари. Малумотлар базаси лойиҳа босқичлари. Моҳият тушунчаси. Алоқа модели тушунчалари. Реляцион маълумотлар базаси дизайни. Маълумотларни моделлаштириш учун бир қанча хилдаги UML нинг бази қисмлари. Калитлар ва функционал боғлиқликлар. Функционал боғлиқликларни ишлатиш орқали декомпозициялаш усуллари. Моҳият алоқа модели. Лойиҳалашда қарор қабул қилиш усуллари.

4 – мавзу: Сўровлар ва уларни қайта ишлаш босқичлари (2 соат).

Сўровларни қайта ишлашнинг асосий принциплари, таҳлил ва таржима, оптималлаштириш, баҳолаш масалари ўрганилади. Сўровларни реляцион алгебра амаллари билан ифодалаш. Сўровларни баҳолаш режаларини ишлаб чиқиш ва режани ўзини баҳолаш. Сўровларни ўлчов бирликлари. Танлаш амалининг муҳимлиги. Танлаш орқали излаш алгоритмлари ва баҳолаш усуллари. Саралаш алгоритмлари ва уларнинг қўлланишлари. Бирлаштириш алгоритмлари.

5 – мавзу: Транзакция. Транзакция концепцияси ва модели (2 соат).

Транзакция тушунчаси, жамиятдаги ўрни ва вазифаси. Транзакция концепцияси. Дастурий таъминот ва МВсини боғлаш ишлари. Транзакцияларнинг хусусиятлари ва уларни бошқариш усуллари. Транзакциянинг моделлари. Транзакция орқали сақлаш моделлари. Қўшилган транзакциялари ва уларни ишлатиш принциплари. Транзакция билан ишловчи SQL операторлари.

6 – мавзу: Маълумотларни йиғиш, жамлаш ва излаш усуллари (2 соат).

Маълумотларни жамлаш ҳақида тушунча. Маълумотларни йиғиш тизимини ишлаб чиқиш. Қарор қабул қилиш масалари ва муаммолари. SQL ва ёрдамчи пакетларнинг вазифалари. Маълумотларни сақлашни ташкиллаштириш масалари ва элементлари. Қоидалар омбори, ечимлар шажарасини шакллантириш масалари ва муаммолари. Энг яхши бўлинишлар алгоритмлари. Маълумотларни излаш учун индекслаш, параметрли ва зонали индекслаш. Долзарбликнинг вазни зоналари. Қийматни машина ўқуви ёрдамида аниқлаш. g нинг энг самарали қиймати. Термин частотаси ва вазни. Тескари маълумотли частота каби излаш усуллари.

АМАЛИЙ МАШҒУЛОТЛАР МАЗМУНИ

1-амалий машғулот (4 соат). SQL тилида маълумотлар базасини яратиш

Университет SQL маълумотлар базаси. Data-definition language. View definition. SQL Data Definition. SQL. Data inconsistency. Consistency constraints. Data abstraction нинг имкониятлари. SQL тилида Университет реализацион модели асосида маълумотлар базасини яратиш учун SQL тилида жадвал ва унинг элементларини шакллантириш. Университет маълумотлар базасини SQL тилида яратиш.

2-амалий машғулот (4 соат). SQL тилида ER диаграммасини яратиш

SQL тилида ER диаграммалар куриш ва уларни ташкиллаштириш учун кўрсатмалар ишлаб чиқиш. Университет ER диаграммалари куриш. Университет ER диаграммаларининг элементлари тушунчалари ва яратиш босқичлари.

3-амалий машғулот (2 соат). SQL тили асосида сўровларни амалга ошириш

SQL оператори ёрдамида танлаш амали орқали сўровларни амалга ошириш. Танлаш орқали излаш алгоритмларига асосланган сўровларни яратиш усуллари. Саралаш алгоритмлари ва уларнинг қўлланишлари. Бирлаштириш алгоритмлари асосида сўровларни қайта ишлаш.

4-амалий машғулот (2 соат). SQL тили асосида транзакцияни яратиш

Маълумотлар базаси учун оддий транзакция яратиш ва тестдан ўтказиш. Транзакциянинг моделлари, транзакция орқали сақлаш моделлари. Қўшилган транзакциялар ва уларни ишлатиш принциплари. Транзакция билан ишловчи SQL операторлари билан ишлаш.

5-амалий машғулот (2 соат). SQL тили асосида маълумотларни йиғиш, жамлаш ва излаш усуллари.

SQL тили асосида маълумотларни йиғиш, жамлаш ва излаш усулларида фойдаланиш.

ЎҚИТИШ ШАКЛЛАРИ

Мазкур модул бўйича қуйидаги ўқитиш шаклларидан фойдаланилади:

- маърузалар, амалий машғулотлар (маълумотлар ва технологияларни англаб олиш, ақлий қизиқишни ривожлантириш, назарий билимларни мустаҳкамлаш);
- давра суҳбатлари (қўрилаётган лойиҳа ечимлари бўйича таклиф бериш қобилиятини ошириш, эшитиш, идрок қилиш ва мантиқий хулосалар чиқариш);
- баҳс ва мунозаралар (лойиҳалар ечими бўйича далиллар ва асосли аргументларни тақдим қилиш, эшитиш ва муаммолар ечимини топиш қобилиятини ривожлантириш).

II БЎЛИМ

ФОЙДАЛАНИЛАДИГАН
ИНТЕРФАОЛ ТАЪЛИМ
МЕТОДЛАРИ

II. ФЙДАЛАНИЛАДИГАН ИНТЕРФАОЛ ТАЪЛИМ МЕТОДЛАРИ.

«Блум кубиги» методи

Методнинг мақсади: Мазкур метод тингловчиларда янги ахборотлар тизимини қабул қилиш ва билимларни ўзлаштирилишини енгиллаштириш мақсадида қўлланилади, шунингдек, бу метод тингловчилар учун “Очиқ” саволлар тузиш ва уларга жавоб топиш машқи вазифасини белгилайди.

Методни амалга ошириш тартиби:

1. Ушбу методни қўллаш учун, оддий куб керак бўлади. Кубнинг ҳар бир томонида қўйидаги сўзлар ёзилади:
 - **Санаб беринг, таъриф беринг (оддий савол)**
 - **Нима учун (сабаб-оқибатни аниқлаштирувчи савол)**
 - **Тушинтириб беринг (муаммони ҳар томонлама қараш саволи)**
 - **Таклиф беринг (амалиёт билан боғлиқ савол)**
 - **Мисол келтиринг (ижодкорликни ривожлантирувчи савол)**
 - **Фикр беринг (таҳлил қилиш ва баҳолаш саволи)**
2. Ўқитувчи мавзуни белгилаб беради.
3. Ўқитувчи кубикни столга ташлайди. Қайси сўз чиқса, унга тегишли саволни беради.

“KWLH” методи

Методнинг мақсади: Мазкур метод тингловчиларда янги ахборотлар тизимини қабул қилиш ва билимларни тизимлаштириш мақсадида қўлланилади, шунингдек, бу метод тингловчилар учун мавзу бўйича қўйидаги жадвалда берилган саволларга жавоб топиш машқи вазифасини белгилайди.

Изоҳ. KWLH:

Know – нималарни биламан?

Want – нимани билишни хоҳлайман?

How - қандай билиб олсам бўлади?

Learn - нимани ўрганиб олдим?.

“KWL” методи	
1. Нималарни биламан: -	2. Нималарни билишни хоҳлайман, нималарни билишим керак: -
3. Қандай қилиб билиб ва топиб оламан: -	4. Нималарни билиб олдим: -

“5W1H” методи

Методнинг мақсади: Мазкур метод тингловчиларда янги ахборотлар тизимини қабул қилиш ва билимларни тизимлаштириш мақсадида қўлланилади, шунингдек, бу метод тингловчилар учун мавзу бўйича қўйидаги жадвалда берилган олти саволларга жавоб топиш машқи вазифасини белгилайди.

What?	Нима? (таърифи, мазмуни, нима учун ишлатилади)	
Where?	Қаерда (жойлашган, қаердан олиш мукин)?	
What kind?	Қандай? (параметрлари, турлари мавжуд)	
When?	Қачон? (ишлатилади)	
Why?	Нима учун? (ишлатилади)	
How?	Қандай қилиб? (яратилади, сақланади, тўлдирилади, таҳрирлаш мумкин)	

“SWOT-таҳлил” методи.

Методнинг мақсади: мавжуд назарий билимлар ва амалий тажрибаларни таҳлил қилиш, таққослаш орқали муаммони ҳал этиш йўлларини топишга, билимларни мустаҳкамлаш, такрорлаш, баҳолашга, мустақил, танқидий фикрлашни, ностандарт тафаккурни шакллантиришга хизмат қилади.

S – (strength)	• кучли томонлари
W – (weakness)	• заиф, кучсиз томонлари
O – (opportunity)	• имкониятлари
T – (threat)	• хавфлар

“БЕЕР” методи

Методнинг мақсади: Бу метод мураккаб, кўптармоқли, мумкин қадар, муаммоли характеридаги мавзуларни ўрганишга қаратилган. Методнинг моҳияти шундан иборатки, бунда мавзунинг турли тармоқлари бўйича бир хил ахборот берилади ва айти пайтда, уларнинг ҳар бири алоҳида аспектларда муҳокама этилади. Масалан, муаммо ижобий ва салбий томонлари, афзаллик, фазилат ва камчиликлари, фойда ва зарарлари бўйича ўрганилади. Бу интерфаол метод танқидий, таҳлилий, аниқ мантикий фикрлашни муваффақиятли ривожлантиришга ҳамда ўқувчиларнинг мустақил ғоялари, фикрларини ёзма ва оғзаки шаклда тизимли баён этиш, ҳимоя қилишга имконият яратади. “Бееp” методидан маъруза машғулотларида индивидуал ва жуфтликлардаги иш шаклида, амалий ва семинар машғулотларида кичик гуруҳлардаги иш шаклида мавзу юзасидан билимларни мустаҳкамлаш, таҳлил қилиш ва таққослаш мақсадида фойдаланиш мумкин.

Методни амалга ошириш тартиби:



тренер-ўқитувчи иштирокчиларни 5-6 кишидан иборат кичик гуруҳларга ажратади;



тренинг мақсади, шартлари ва тартиби билан иштирокчиларни таништиргач, ҳар бир гуруҳга умумий муаммони таҳлил қилиниши зарур бўлган қисмлари туширилган тарқатма материалларни тарқатади;



ҳар бир гуруҳ ўзига берилган муаммони атрофлича таҳлил қилиб, ўз мулоҳазаларини тавсия этилаётган схема бўйича тарқатмага ёзма баён қилади;



навбатдаги босқичда барча гуруҳлар ўз тақдимотларини ўтказадилар. Шундан сўнг, тренер томонидан таҳлиллар умумлаштирилади, зарурий ахборотлар билан тўлдирилади ва мавзу яқунланади.

Муаммоли савол					
1-усул		2-усул		3-усул	
афзаллиги	камчилиги	афзаллиги	камчилиги	афзаллиги	камчилиги
Хулоса:					

“Кейс-стади” методи

«Кейс-стади» - инглизча сўз бўлиб, («case» – аниқ вазият, ҳодиса, «stady» – ўрганмоқ, таҳлил қилмоқ) аниқ вазиятларни ўрганиш, таҳлил қилиш асосида ўқитишни амалга оширишга қаратилган метод ҳисобланади. Мазкур метод дастлаб 1921 йил Гарвард университетиде амалий вазиятлардан иқтисодий бошқарув фанларини ўрганишда фойдаланиш тартибида қўлланилган. Кейсда очик ахборотлардан ёки аниқ воқеа-ҳодисадан вазият сифатида таҳлил учун фойдаланиш мумкин.

“Кейс методи” ни амалга ошириш босқичлари

Иш босқичлари	Фаолият шакли ва мазмуни
1-босқич: Кейс ва унинг ахборот таъминоти билан таништириш	<ul style="list-style-type: none"> ✓ якка тартибдаги аудио-визуал иш; ✓ кейс билан танишиш(матнли, аудио ёки медиа шаклда); ✓ ахборотни умумлаштириш; ✓ ахборот таҳлили; ✓ муаммоларни аниқлаш
2-босқич: Кейсни аниқлаштириш ва ўқув топшириғни белгилаш	<ul style="list-style-type: none"> ✓ индивидуал ва гуруҳда ишлаш; ✓ муаммоларни долзарблик иерархиясини аниқлаш; ✓ асосий муаммоли вазиятни белгилаш
3-босқич: Кейсдаги асосий муаммони таҳлил этиш орқали ўқув топшириғининг ечимини излаш, ҳал этиш йўллари ишлаб чиқиш	<ul style="list-style-type: none"> ✓ индивидуал ва гуруҳда ишлаш; ✓ муқобил ечим йўллари ишлаб чиқиш; ✓ ҳар бир ечимнинг имкониятлари ва тўсиқларни таҳлил қилиш; ✓ муқобил ечимларни танлаш
4-босқич: Кейс ечимини ечимини шакллантириш ва асослаш, тақдимот.	<ul style="list-style-type: none"> ✓ якка ва гуруҳда ишлаш; ✓ муқобил вариантларни амалда қўллаш имкониятларини асослаш; ✓ ижодий-лойиҳа тақдимотини тайёрлаш; ✓ якуний хулоса ва вазият ечимининг амалий аспектларини ёритиш

“Ассесмент” методи

Методнинг мақсади: мазкур метод таълим олувчиларнинг билим даражасини баҳолаш, назорат қилиш, ўзлаштириш кўрсаткичи ва амалий кўникмаларини текширишга йўналтирилган. Мазкур техника орқали таълим олувчиларнинг билиш фаолияти турли йўналишлар (тест, амалий кўникмалар, муаммоли вазиятлар машқи, қиёсий таҳлил, симптомларни аниқлаш) бўйича ташҳис қилинади ва баҳоланади.

Методни амалга ошириш тартиби:

“Ассесмент”лардан маъруза машғулотларида талабаларнинг ёки қатнашчиларнинг мавжуд билим даражасини ўрганишда, янги маълумотларни баён қилишда, семинар, амалий машғулотларда эса мавзу ёки маълумотларни ўзлаштириш даражасини баҳолаш, шунингдек, ўз-ўзини баҳолаш мақсадида индивидуал шаклда фойдаланиш тавсия этилади. Шунингдек, ўқитувчининг ижодий ёндашуви ҳамда ўқув мақсадларидан келиб чиқиб, ассесментга қўшимча топшириқларни киритиш мумкин.

Ҳар бир катакдаги тўғри жавоб 5 балл ёки 1-5 балгача баҳоланиши мумкин.



Тест



Муаммоли вазият



**Тушунча таҳлили
(симптом)**



Амалий вазифа

“Инсерт” методи

Методни амалга ошириш тартиби:

- ўқитувчи машғулотга қадар мавзунинг асосий тушунчалари мазмуни ёритилган матнни тарқатма ёки тақдимот кўринишида тайёрлайди;
- янги мавзу моҳиятини ёритувчи матн таълим олувчиларга тарқатилади ёки тақдимот кўринишида намойиш этилади;
- таълим олувчилар индивидуал тарзда матн билан танишиб чиқиб, ўз шахсий қарашларини махсус белгилар орқали ифодалядилар. Матн билан ишлашда талабалар ёки қатнашчиларга қуйидаги махсус белгилардан фойдаланиш тавсия этилади:

Белгилар	Матн
“V” – таниш маълумот.	
“?” – мазкур маълумотни тушунмадим, изоҳ керак.	
“+” бу маълумот мен учун янгилик.	
“– ” бу фикр ёки мазкур маълумотга қаршиман?	

Белгиланган вақт якунлангач, таълим олувчилар учун нотаниш ва тушунарсиз бўлган маълумотлар ўқитувчи томонидан таҳлил қилиниб, изоҳланади, уларнинг моҳияти тўлиқ ёритилади. Саволларга жавоб берилади ва машғулот якунланади.

Ш БЎЛИМ

НАЗАРИЙ
МАТЕРИАЛЛАР

III. НАЗАРИЙ МАТЕРИАЛЛАР

1 - мавзу: Маълумотлар ва уларнинг турлари. Маълумотлар базаси архитектураси (2 соат)

Режа:

- 1.1. Маълумотлар базасига кириш.
- 1.2. Маълумотларнинг кўриниши ва турлари.
- 1.3. Маълумотлар базасининг архитектураси.
- 1.4. Реляцион моделга тушунчаси.

Таянч иборалар: *маълумотлар, архитектура, модел, реляцион модел, Маълумотлар базаси, Маълумотлар базасини бошқариш тизимлари.*

1.1. Маълумотлар базасига кириш

Маълумотлар базасини бошқариш тизими – ўзаро боғланган маълумотлар тўплами ва бу маълумотларга боғланишга ёрдам берувчи бириктирилган дастурлар. Бу маълумотлар тўплами маълумотлар базаси дейилади ва у ўз ичига бирор корхона ёки иш фаолиятнинг муҳим ва фойдали маълумотларини олади. Маълумотлар базасини бошқариш тизимининг асосий мақсади - маълумотлар базаси маълумотларини сақлаш ва маълумотлардан қулай ва самарали тарзда фойдаланишни таъминлаш.

Маълумотлар базаси тизими катта ҳажмдаги маълумотларни бошқариш учун тузилган. Маълумотларни бошқариш иккита нарсани талаб қилади:

- маълумотларни сақлаш учун уларни структурасини аниқлаш;
- маълумотларни бошқариш учун тизимни таъминлаш.

Шунингдек маълумотлар базаси тизими сақланган маълумотларни хавфсизлигини тизим зарарланишидан, рухсатсиз киришга уринилган сай-ҳаракатлардан сақлаши лозим.

Агар маълумот бир неча фойдаланувчилар ўртасида тақсимланган бўлса, тизим бўлиши мумкин бўлган нотўғри натижалардан қутулиши лозим. Чунки маълумот жуда кўп корхоналарда муҳим ҳисобланади. Компютер олимлари маълумотларни бошқариш учун бир қанча техникавий ва ғоявий улулар билан ривожлантиришди. Бу боб маълумотлар базаси тизимининг тамойиллари билан қисқача таништиради.

Маълумотлар базасидан кенг фойдаланилади. Қуйида бази қўлланиш соҳалари келтирилади:

Ташкилот маълумотлари

-чегирмалар: харидорлар учун, маҳсулотлар харид қилинган товарлар маълумотлари;

-ҳисобчилик иши: тўловлар учун, кирим ва чиқимлар ва бошқа ҳисоб-

китоб маълумотлари;

-инсон ресурслари: ишчилар ҳақида маълумотлар, иш ҳақлари, тўлов солиқлари, фойда ва бошқ.;

-унверситетлар: талабалар маълумотлар, йўналишлар рўйхати ва бошқа маълумотлар.

Бугунги кунда маълумотлар базаси ҳар бир ташкилот корхонанинг асосий қисми. Дастлаб жуда кам одамлар маълумотлар базаси тизими билан ишлашган. Улар маълумотлар базасини яхши тушунишмаган. Кейинчалик бу тизим жуда кўп соҳаларда қўлланила бошланган. Мисол учун банк соҳасида, ҳаво линеарларида ва бошқа кўплаб автоматик тизимларда¹.

1990-йилларнинг охиридан бошлаб интернет кескин ривожана бошлади. Ташкилотлар маълумотлар базаси билан мулоқот ойнасини ёб интерфасе билан алмаштиришди ва кўплаб тўғридан –тўғри бажаришга имкон берувчи хизматларни таклиф қилишди. Мисол учун сиз ёб браузерда бирор китоблар тўпламидан қайсидир китобни ўқимоқчи бўлсангиз, сиз тўғридан –тўғри китоблар маълумотлар базасига

Мурожаат қиласиз ва боғланасиз ёки сиз бирор банкнинг вебсайтига кириб у ҳақида маълумот олмақчи бўлсангиз, сиз тўғридан-тўғри шу банкнинг маълумотлар тизимига боғланасиз ва маълумотларни ўша едан оласиз.

Маълумотлар базаси тизимнинг муҳимлигини бугунда бошқа йўл билан ҳам баҳолаш мумкин. Мисол учун Oracle компанияси дунёда йирик дастурий маҳсулот ишлаб чиқарувчи компания ҳисобланади ва бу компания маълумотлар базасини ишлаб чиқиш, ривожлантириш ва сотиш билан шуғулланади, шунингдек маълумотлар базаси Microsoft ва IBM компаниялари дастурий маҳсулотларининг муҳим қисмини ташкил қилади.

1.2. Маълумотларнинг кўриниши ва турлари².

Маълумотлар базаси тизими –ўзаро боғланган маълумотлар тўплами ва бу маълумотлардан фойдаланиш имконини берувчи дастурдан иборат. Маълумотлар базаси тизимининг асосий мақсади - фойдаланувчилар маълумотларини абстракт кўриниши билан таъминлаш. Тизим маълумотларни қандай сақланганлиги ва ҳимояланганлигини яширади.

Тизим учун керакли бўлган маълумотлар тезда базадан етказилиши керак. Самарадорликни ошириш ва маълумотлар базасидаги маълумотларни кўрсатиш учун яратувчилар мураккаб маълумот тузилмасидан фойдаланган.

¹ Silberschatz, Abraham. Database system concepts // pages 1-2

² Silberschatz, Abraham. Database system concepts // page 6

Тузувчилар бу мураккаб жараённинг бир неча босқичларини фойдаланувчиларга тизимдан фойдаланишни осонлаштириш мақсадида яширишган. Бу босқичлар куйидагилар бўлган:

- **Физик босқич.** Бу босқич маълумотларни қандай сақланганлиги, уларнинг тузулиш тавсилотларини тасвирловчи энг паст мураккаб босқич ҳисобланади.

- **Мантикий босқич.** Бу босқич маълумотлар базасидаги маълумотларнинг ўзаро қандай боғланганлиги ва уларнинг маълумотлар базасида қандай сақланганлигини кўрсатади. Бу босқич ишлашида физик босқичга мурожат қилади. Маълумотлар базаси бошқарувчиси шу босқичда иш кўради.

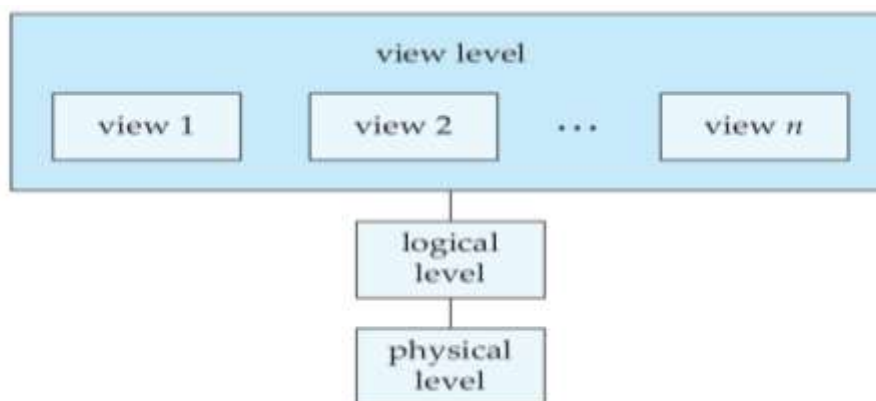
- **Кўрсатиш босқичи.** Бу босқич бутун бир маълумотлар базасидан фақат бир қисмини кўрсатди. Базадаги маълумотлар жуда кўп бўлади, фойдаланувчи эса улардан фақат ўзига кераклисини кўриши лозим бўлади. Шунинг учун ҳам бу босқич ишлаб чиқилган. Тизим айна бир маълумотлар базаси учун кўпгина кўрсатишларни таъминлаши мумкин.

Кўп юқори дастурлаш тиллари маълумотларнинг бундай тузулиш турини қўллаб қувватлайди³.

Мисол учун

```
Type instructor = record
  Id : char(5);
  Name : char(20);
  Salary : numeric(8,2);
end;
```

Бу код ўзида 3 та ёзув майдони бўлган рўйхатни билдиради. Шунингдек маълумотлар базасидаги маълумотлар ўзгартирилиши, ўчирилиши ёки янги маълумот қўшилиши мумкин.



1.1- расм. Маълумотлар тузилишининг 3 та босқичи.

³ 1. Silberschatz, Abraham. Database system concepts // 6 page

Маълумот моделлари⁴

Маълумотлар базаси тузилмасининг асоси бу маълумот моделларидир. У ўз ичига маълумотларнинг ўзаро боғланиши, уларнинг тузулиш кетма-кетлиги ва бошқа бир қанча маълумотларни олади. Модел маълумотлар базасида физик, мантикий ва кўрсатиш босқичларини тасвирлашни ўзига хос бир йўли. Маълумот моделлари қуйидаги 4 та туркумга бўлинади:

• **Муносабат модели.** Жадвалларни ва улардаги маълумотларни ўзаро боғланиш муносабатларини тасвирлаш учун фойдаланилади. Ҳар бир жадвалнинг бир неча устунлари ва ҳар бир устуннинг ягона такрорланмас номи бўлади. Бу модел маълумотлар базасини бошқариш тизимида жуда кенг фойдаланилади.

• **Моҳият - алоқа модели.** Бу моделдан асосан объектларнинг йиғиндисини ва улар ўртасидаги боғлиқликни тасвирлайди. Моҳият бу бирор нарса ёки объект ҳақиқий дунёдаги бошқа объектлардан фарқлайди. Бу моделдан маълумотлар базаси тузилмасида кенг фойдаланилади.

• **Объект-база маълумот модели.** Объектга йўналтирилган дастурлаш дастурлаш методлари ичида устувор ҳисобланади. Объектга йўналтирилган маълумот моделида тайёр объектлардан фойдаланилади.

• **Яримтузилган маълумот модели.** Бу модел маълумотларнинг ўзига хос хусусиятларини аниқлашга имкон беради. Бу модел юқорида келтириб ўтилган моделларга зид ишлайди. Бунда ҳар бир маълумот қисмини ўзига хос хусусияти ўрнатилган бўлиши керак.

SQL (Structured Query Language) Структуралашган сўровлар тили. Аниқ қилиб айтганда SQL – маълумотларни қисм тили, чунки Маълумотлар базасини бошқариш тизими бошқа тил воситаларига ҳам эга. SQL яратилаётганда, оддий табиий тилга яқин реляцион маълумотлар базасини сўровлар тили сифатида лойиҳаланган.

SQL тили декларатив (тавсифловчи) тил ҳисобланади, шу билан у дастурлаш тилларидан фарқ қилади. Буни маъноси SQL тилида бериладиган ифодалар нима қилиш кераклигини тавсифлайди, яъни уни қандай бажариш кераклигини кўрсатмайди.

SQL тили:

- Операторлар;
- Кўрсатмалар;
- Функциялар;

тўпламидан иборат. Сўровлар яратишни баъзи қоидалари:

SQL тилида ифодаларда калит сўзлар кичик ва катта ҳарфлар билан

⁴ 1.Silberschatz, Abraham. Database system concepts // 8-9 pages

ёзилиши мумкин. Ифодалар битта ёки бир нечта сатрда ёзилиши мумкин. Ифода якунида (охирода) «;» белги қўйилади. Агар SQL командасини бир нечта блоки ёзилса, улар бир биридан “;” билан ажратилади.

Барча SQLни калит сўзлари (командалари, операторлари ва бошқалар) фақат хизматчи сўз сифатида ишлатилиши мумкин, бошқа мақсадларда яъни жадвал номи, устун номи, ўзгарувчи ва бошқалар сифатида ишлатиш мумкин эмас.

SQL тили эркин форматга эга бўлгани учун, SQL алохида операторлари ва уларнинг кетма-кетлигини, алохида ажратиб ёзиш ва текислаб ёзишни ишлатиш мумкин. Қуйидаги қоидаларга бўйсунтириш талаб этилади:

- ✓ оператордаги ҳар бир конструкция янги сатрдан бошланиши керак;
- ✓ ҳар бир конструкция бошланишида ташлаб кетладиган бўш позициялар, бошқа оператор конструкциялари ҳам бўлиши керак;
- ✓ агар конструкция бир неча қисмдан иборат бўлса, уларнинг ҳар бири қисм янги сатрлардан бўш ўринларни олдинги конструкцияга нисбатан силжитиб ёзилади.

SQL операторларини 4 та гуруҳга бўлиш мумкин:

- DML (Data Manipulation Language) – маълумотлар билан манипуляциялаш операторлари, маълумотлар базаси билан ишлашни таъминлашга мўлжалланган тил ҳисобланади;
- DDL (Data Definition Language) – маълумотларни аниқлаш операторлари. DDL операторлар гуруҳи маълумот база объектларини яратиш, такомиллаштириш ва олиб ташлаш учун мўлжалланган;
- DCL (Data Control Language) - маълумотларни бошқариш тили. Бу тил маълумотлар базасини ҳар хил ўринсиз зарарланишлардан сақланиши таъминлаш мўлжалланган;
- TCL (Transaction Control Language) - транзакцияларни бошқарув тили. Бу гуруҳ операторлари DML гуруҳ операторлари бажарган операцияларни ўзгаришини таъминлайди;

SQL тилида маълумотлар билан манипуляция амаллари учун тўртда калит сўз ишлатилади: SELECT, INSERT, UPDATE, DELETE. SELECT операторли ифодалар алохида ўрин эгаллайди, чунки у маълумотлар ажратиб олиш учун мўлжалланган ва бу фойдаланувчилар ечадиган масалаларни кўп қисмини ташкил қилади. SELECT оператори кўрсатилган ажратиш мезонига мос равишда бир ёки бир неча жадвалдан атрибутларни ажратиш учун ишлатилади.

SELECT (танлаш) SQL тилининг энг муҳим ва кўп ишлатиладиган оператори ҳисобланади. У маълумотлар базаси жадвалидан ахборотларни

танлаб олиш учун мўлжалланган.

SELECT оператори ни умумий кўриниши куйидагича:

SELECT [**DISTINCT**] < *атрибутлар руйхати* >

FROM < *жадваллар руйхати* >

[**WHERE** < *танлаш шарти* >]

[**ORDER BY** < *атрибутлар руйхати* >]

[**GROUP BY** < *атрибутлар руйхати* >]

[**HAVING** < *шарт* >]

[**UNION** < **SELECT** *операторли ифода*>] ;

Бу ерда квадрат қавсларда ёзилган элементлар хар доим хам ёзилиши шарт эмас.

DDL Операторлари:

- CREATE SCHEMA - Маълумотлар базаси схемасини яратиш;
- DROP SHEMA - Маълумотлар базаси схемасини ўчириш;
- CREATE TABLE - Жадвал яратиш;
- ALTER TABLE - Жадвални ўзгартириш;
- DROP TABLE - Жадвални ўчириш;
- CREATE DOMAIN - Домен яратиш;
- ALTER DOMAIN - Доменни ўзгартириш;
- DROP DOMAIN - Доменни ўчириш;
- CREATE COLLATION - Кетма-кетлик яратиш;
- DROP COLLATION - Кетма-кетликни ўчириш;
- CREATE VIEW - Тасаввур яратиш;
- DROP VIEW - Тасаввур ўчириш;

Жадвал **CREATE TABLE** командаси орқали яратилади. Бу команда бўш (empty), яъни қортежларсиз (қаторларсиз) жадвал яратади:

CREATE TABLE < жадвал-номи >

(< майдон номи > < майдон тоифаси >[(<размери>)],

< майдон номи > < майдон тоифаси >[(<размери>)],);

Устунлар тартиби уларни кўрсатилган тартибида белгиланади. Майдон номи янги қаторга ўтказиб ёзилиши мумкин эмас, лекин улар", "билан ажратилади.

CREATE TABLE PEOPLE (

ID **NUMBER**,

NM **VARCHAR2**(50),

FAMIL **VARCHAR2**(50),

OTCH **VARCHAR2**(50),

DROG **DATE**)

Ҳозирги кунда асосан реляцион маълумотлар базасига асосланган тизимлар кўп ишлатилмоқда. Унча катта бўлмаган ташкилотларда

Маълумотлар базасини бошқариш тизими битта компьютерда юклаб ишлатиш мумкин. Бундай шахсий маълумот база тизими бир нечта фойдаланувчи ишлаганда фойдали бўлиши мумкин. Шунинг учун hozirги кунда корхона ва ташкилотларда маълумот базаси бирламчи серверда жойлаштирилади, фойдаланувчилар эса алоҳида компьютерлар билан уланади. Бу компьютерларни мижоз (клиент) компьютерлари дейилади. Сервер маълумотларни марказлашган сақловчиси ҳисобланади, шунинг учун барча фойдаланувчилар бир ахборотларни, жумладан ўзгартирилган ахборотларни кўриш имкониятига эга бўлади.

Бундай жадвалларни қуйидаги хусусиятлари бўлади.

- Жадвални ҳар бир элементи битта маълумот элементи ҳисобланади ва такрорланувчи гуруҳлар бўлмайди;
- Барча устунлар бир жинслидир ёки устунлари бир тоифада бўлади яъни элементлар бир хил табиатга эга;
- Ҳар бир устунга ном тайинланган;
- Жадвалда бир хил сатр 2 марта учрамайди;
- Бундай жадваллар билан амал бажарилганда сатр ва устунлар ихтиёрий тартибда қаралади ва ихтиёрий кетма-кетликда ишлатилиши мумкин.

ID	name	dept_name	salary
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califeri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

a) Instructor jadvali

dept_name	building	budget
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

b) Jadvalning bir qismi.

1.2-расм. Маълумотлар⁵

Транзакцияларни бошқариш⁶

Транзакция – маълумотларни қайта ишлаш операторлар кетма-кетлиги бўлиб, маълумот базаси билан ишлайдиган бўлинмас бирлик ҳисобланади. Транзакция маълумот базасини бир ҳолатдан бошқа ҳолатга ўтказди. Транзакция маълумотларни яхлитлигини сақлаган ва параллел

⁵ Silberschatz, Abraham. Database system concepts // 13 - page

⁶ Silberschatz, Abraham. Database system concepts // 22-23 pages

бажарилаётган транзакцияларга боғлиқ бўлмаган холда тўлиқ ва муваффақиятли бажарилиши ёки умуман бажарилмаслиги мумкин, бу холда ҳеч қандай натижа кузатилмайди. Транзакциялар транзакцион системалар ёрдамида қайта ишланади ва ишлаш жараёнида унинг тарихи сақланиб борилади. Агар транзакция бажариш даврида операцион тизим ёки иловани ишлашида нуқсон (носозлик) учраса, нуқсон тугатилгандан кейин маълумот базасидаги маълумотлар аввалги ўз ҳолатига қайтади. Транзакция табиий яқунлангунча, уни “қайтариш” учун (**ROLLBACK**) командаси ишлатилиши мумкин.

Транзакцияларда ва транзакцион системаларда энг кенг тарқалган хусусиятлари (мажмуалар) тўплами – **ACID**:

- **Atomicity** — АТОМАРЛИК – системада ҳеч қандай транзакция қисман бажарилмаслигини кафолат беради;

- **Consistency** — КЕЛИШИЛГАНЛИК – транзакция бошланмасдан олдин система келишилган ҳолатда бўлади ва шу ҳолатда транзакция яқунлангунга қадар қолиши шарт;

- **Isolation** — ЯККАЛИК – транзакция бажарилиш жараёнида параллел бажарилаётган бошқа транзакциялар унинг натижасига таъсир қилмайди;

- **Durability** – МУСТАХКАМЛИК – қуйи босқичдаги муаммоларга қарамасдан муваффақиятли тугалланган транзакция система ҳолати тиклангандан сўнг сақланган холда бўлиши зарур;

Транзакцияни фиксация қилиш ёки орқага қайтариш учун SQL - тилида **COMMIT [WORK]**, **SAVEPOINT** ва **ROLLBACK [WORK]** командалар ишлатилади.

COMMIT [WORK] – транзакцияни фиксирлайди. У қуйидаларни бажаришни таъминлайди:

- Жорий транзакция қилинган барча ўзгаришларни фиксирлайди;
- Транзакция тугалланади;
- Ушбу транзакция учун барча нуқталар йўқотилади;
- Транзакция жараёнида ишлатилган барча объектлар бўшатилади.
- **COMMIT [WORK]** командасини ишлаш сеанси тугаганда ҳам бериш тавсия қилинади.

ROLLBACK [WORK] команда маълумот базасида транзакцияни қайтаришни, яъни ўзгаришларни бекор қилинишини англатади. Орқага қайтиш одатда иловада хато бўлганда, маълумотларни яхлитлиги бузилганда ва х.к.з. холларда бажарилади. Уни синтаксиси қуйидагича

ROLLBACK [WORK] TO сақлаш_нуқтасини_номи;

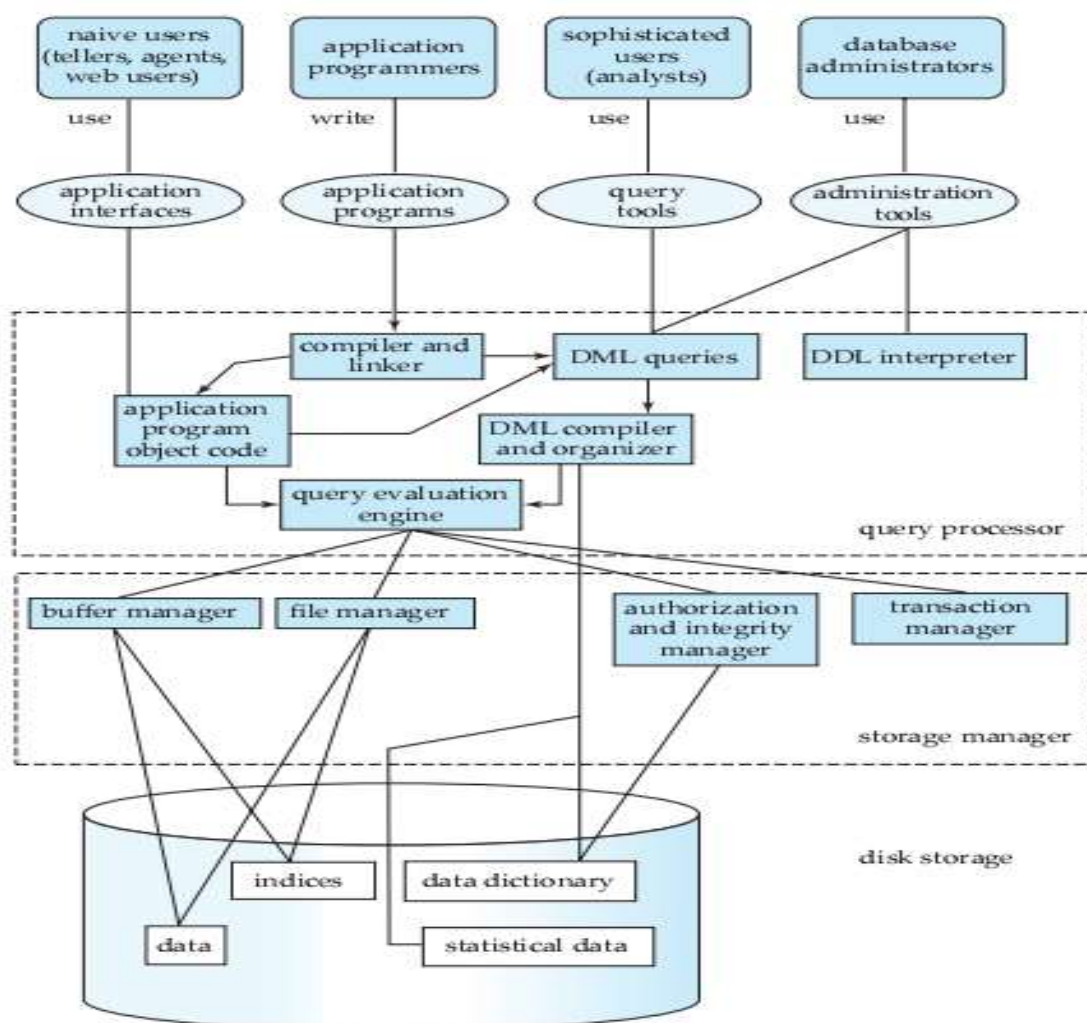
Уни ишлатиш қуйидаги амалларни бажаришга олиб келади.

- ✓ транзакцияни бажарилиши тугатилади;
- ✓ жорий транзакцияда бажарилган барча ўзгаришлар бекор қилинади;
- ✓ транзакциядаги барча блокировкалар бекор қилинади.

1.3. Маълумотлар базасининг архитектураси⁷

Маълумотлар базаси тизимининг архитектурасига у ишлаётган компьютер тизими жуда катта таъсир қилади. Маълумотлар базаси тизими марказлашган ёки клиент-сервер технологияси асосида ишлай олади. Клиент-сервер технологиясида сервер бирор бошқа жойда бошқа бирор компьютерда ишлаши мумкин. Бу технология орқали битта сервердан кўплаб клиентлар фойдаланиши мумкин. Улар бошқа жойдаги серверга асосан тармоқ орқали уланади. Фойдаланувчилар маълумотлар базасидаги маълумотларни ўзгартириши, қўшиши ёки ўчириши мумкин, шунингдек улар бошқа фойдаланувчилар қилган ўзгартиришларни ҳам кўриб боришлари мумкин.

⁷ Silberschatz, Abraham. Database system concepts // 23-24 pages



1.3 – расм. Маълумотлар базаси тизимининг тузилиши.

Юқоридаги расмда маълумотлар базасини бошқариш тизимининг тузилиш ва ишлаш принциплари келтирилган. Яъни маълумотларнинг хотирага қандай жойлаштирилиши, сўровлар, улар ўртасидаги боғланишлар, транзаксияларни бошқариш ва албатта фойдаланувчилар билан ишлайдиган қулай интерфейсни қандай ишлаши келтириб ўтилган.

1.4. Реляцион моделга кириш⁸

Реляцион модел маълумотларни қайта ишловчи дастурларнинг асосий маълумотлар модели ҳисобланади. Бошқа модел турларига нисбатан содда, дастурчи ишини осонлаштирувчи модел бўлгани учун ҳозирги кунда кенг қўлланилади.

Реляцион маълумотлар базаси структураси

Реляцион маълумотлар базаси ягона номга эга бўлган жадваллардан

⁸ Silberschatz, Abraham. Database system concepts // 39-40 pages

ташқил топади. Мисол учун Инструкторлар жадвалини (1.4- расм) кўриб чиқамиз. Жадвал 4 та устундан иборат ва улар қуйидагича номланган: *ID*, *name*, *dept_name* ва *salary*. Жадвалдаги ҳар бир сатр инструктор ҳақидаги маълумотларни сақлайди. Мос равишда Курслар жадвали (1.5- расм) *source_id*, *title*, *dept_name* ва *credit* деб номланган устунлардан иборат бўлган маълумотларни сақлайди.

Учинчи пререквизит жадвали (1.6- расм) ҳар бир курс учун пререквизит курслар ҳақидаги маълумотларни сақлайди. Жадвал иккита *source_id* ва *prereq_id* устунларидан ташқил топган. Шундай қилиб, кўриниб турибдики пререквизит жадвалидаги маълумотлар ўзаро боғланган. Умуман олганда, жадвалдаги сатр қийматлар мажмуи орасидаги муносабатни ифодалайди. Жадвалдаги бундай муносабатлар ва математик муносабат атамаси орасида ўхшашлик мавжуд бўлгани сабабли бундай модел реляцион деб номланади. Математикада кортеж бу қийматлар рўйхати ёки кетма-кетлиги. N қийматлар орасидаги муносабатлар математик нуқтайи назардан N-кортеж орқали ифодаланади. N қийматли кортеж жадвалнинг сатрига тўғри келади.

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

1.4-расм. Маълумотлар

<i>course_id</i>	<i>title</i>	<i>dept_name</i>	<i>credits</i>
BIO-101	Intro. to Biology	Biology	4
BIO-301	Genetics	Biology	4
BIO-399	Computational Biology	Biology	3
CS-101	Intro. to Computer Science	Comp. Sci.	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3
CS-319	Image Processing	Comp. Sci.	3
CS-347	Database System Concepts	Comp. Sci.	3
EE-181	Intro. to Digital Systems	Elec. Eng.	3
FIN-201	Investment Banking	Finance	3
HIS-351	World History	History	3
MU-199	Music Video Production	Music	3
PHY-101	Physical Principles	Physics	4

1.5-расм. Маълумотлар

<i>course_id</i>	<i>prereq_id</i>
BIO-301	BIO-101
BIO-399	BIO-101
CS-190	CS-101
CS-315	CS-101
CS-319	CS-101
CS-347	CS-101
EE-181	PHY-101

1.6-расм. Маълумотлар

Шундай қилиб, реляцион моделда муносабат тушунчаси жадвални, кортежеса сатрни ифодалайди. Мос равишда атрибут атамаси жадвалдаги устунни ифодалайди. Бу муносабатлар Университетнинг бир қисмини ташкил қилади. Ифодалашни осонлаштириш мақсадида биз Университет маълумотлар базасининг аслини олмаймиз.

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

1.7-расм. Маълумотлар

Муносабатдаги кортежларнинг кетма-кетлиги ҳеч нимага таъсир қилмайди. Чунки муносабат кортежлар рўйхатидан ташкил топади. 1.4-расмдаги сараланган муносабатлар кортежива 1.7- расмдаги сараланмаган муносабатлар кортежи бир хил. Чунки бу жадваллардаги кортежлар рўйхати бир хил. Тушуниш осон бўлиши учун кейинчалик муносабатларни биринчи устуни бўйича саралаймиз. Муносабатдаги ҳар бир атрибут учун рухсат етилган қийматлари мавжуд. Бу қийматлар атрибутнинг домени деб аталади. Шу сабабли, инструктор жадвалидаги “*salary*” атрибути доменида бўш бу атрибут қабул қилиши мумкин бўлган қийматлар мажмуидир.

Барча муносабатлардаги атрибутлар атомар ҳисобланади. Агар доменнинг қийматлари бўлинмайдиган қиймат бўлса домен атомар ҳисобланади. Мисол учун Инструктор жадвалида телефон рақамлари

атрибути мавжуд бўлган бўлсин. Бунда бу муносабат атомар ҳисобланмайди. Чунки бу соҳа телефон рақамлари мажмуидир ва бу мажмуада алоҳида телефон рақамлари мавжуд.

Асосий масала домен емас балки бу домен элементларни маълумотлар базасида ишлатишдир. Тасаввур қиламиз, телефон рақами атрибути битта телефон рақамини сақласин. Шунда ҳам биз уни давлат рақами, шаҳар рақами, маҳаллий номер каби атрибутларга бўлсак ҳам у атомар ҳисобланмайди. Агар ҳар бир телефон рақамини бўлинмайдиган қиймат деб тасаввур қилсак, бу домен атомар ҳисобланади. 3 дан 6 гача бобларда ҳамма атрибутлар атомар деб ҳисоблаймиз.

NULL қиймати нўмалумлик ёки мавжудмасликни билдиради. Юқоридаги мисолни давом эттирган ҳолда инструктор телефон рақами киритилмаган ёки мавжуд бўлмаса, биз ушбу NULL қийматини ушбу атрибутга ўзлаштирамиз. Кейинчалик биз бу қиймат бир қатор муаммоларни юзага чиқаришини ва иложи борича ушбу қийматни камроқ ишлатиш кераклигини кўриб ўтаемиз.

Маълумотлар базаси схемаси⁹

Маълумотлар базаси ҳақида гап борганида биз биринчи навбатда маълумотлар базаси схемаси ва маълумотлар базаси ҳолати ўртасидаги фарқни тушунишимиз керак. Маълумотлар базаси схемаси бу маълумотлар базасини мантиқий ташкил этишдан, маълумотлар базаси ҳолати эса оний вақтдаги маълумотлар тасвиридан иборат.

Умуман олганда муносабатлар схемаси атрибутлар рўйхати ва уларга мос доменлардан иборатдир. Маълумотлар базаси ҳолати дастурлаш тилларининг ўзгарувчилар қийматига мос тушади. Вақт ўтиши билан бу қийматлар ўзгариши мумкин. Аксинча, маълумотлар схемаси эса ўзгармайди.

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Biology	Watson	90000
Comp. Sci.	Taylor	100000
Elec. Eng.	Taylor	85000
Finance	Painter	120000
History	Painter	50000
Music	Packard	80000
Physics	Watson	70000

1.8-расм. Маълумотлар

Департамент (1.8- расм) муносабатини кўриб чиқамиз. Бу муносабат учун муносабат схемаси куйидагича:

department (dept_name, building, budget)

⁹ Silberschatz, Abraham. Database system concepts // 42-43 pages

Аҳамият берсак dept_name инструктор ва департамент схемаларида иштирок этмоқда. Бу қайтарилиш бекорга эмас. Бир хил атрибутларни ишлатиш алоҳида муносабатлар ўртасидаги кортежларни бўлиш учун ишлатилади. Мисол учун бизга Watson биносида ишловчи барча инструкторларни топиш керак бўлсин. Биринчи навбатда департамент муносабатига қараймиз. Бу ердан dept_name атрибутидан Вацон ёзувли кортежларни топамиз. Топилган маълумотмарга қараб инструктор муносабатидаги dept_name атрибутга мос тушувчи маълумотларни топишимиз мумкин. Университет мисоли билан давом этамиз. Университетдаги ҳар бир курс бир неча марта тақдим қилиниши мумкин. Семестрлар орасида ёки семестер пайтида. Бизга ушбу ҳар бир тақдимни муносабат орқали ифодалашимиз керак. Схема қуйидагича бўлади:

section (course_id, sec_id, semester, year, building, room_number, time_slot_id)

course_id	sec_id	semester	year	building	room_number	time_slot_id
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

1.9-расм. Маълумотлар

Инструкторлар ва синф бўлимлари орасидаги муносабатни акс эттириш учун қуйидаги муносабат ўринли:

teaches (ID, course_id, sec_id, semester, year)

<i>ID</i>	<i>course_id</i>	<i>sec_id</i>	<i>semester</i>	<i>year</i>
10101	CS-101	1	Fall	2009
10101	CS-315	1	Spring	2010
10101	CS-347	1	Fall	2009
12121	FIN-201	1	Spring	2010
15151	MU-199	1	Spring	2010
22222	PHY-101	1	Fall	2009
32343	HIS-351	1	Spring	2010
45565	CS-101	1	Spring	2010
45565	CS-319	1	Spring	2010
76766	BIO-101	1	Summer	2009
76766	BIO-301	1	Summer	2010
83821	CS-190	1	Spring	2009
83821	CS-190	2	Spring	2009
83821	CS-319	2	Spring	2010
98345	EE-181	1	Spring	2009

1.10-расм. Маълумотлар

Кўриниб турибдики реал университет маълумотлар базасида бундан кўпроқ муносабатлар мавжуд. Юқоридаги муносабатларга қўшимча яна куйидаги муносабатларни ишлатамиз:

- *student (ID, name, dept_name, tot_cred)*
- *advisor (s_id, i_id)*
- *takes (ID, course_id, sec_id, semester, year, grade)*
- *classroom (building, room_number, capacity)*
- *time slot (time_slot_id, day, start_time, end_time)*

Калитлар¹⁰

Берилган муносабатлар ўртасидаги алоқаларни қуриш йўлини кўриш лозим. Бу атрибут қийматлари орасидан шундай ягона қиймат бўлиши керакки, у кортежнинг қийматини ўзида ифода этади. Супер калит бу муносабатдаги ягона (уникал) кортежни аниқловчи қийматдир. Мисол учун инструктор муносабатидаги ID атрибути ҳар бир кортеждан фарқ қилади. Шу сабабли ID супер калит ҳисобланади. Name атрибути эса супер калит бўла олмайди. Чунки name атрибути қиймати бошқа бир кортежда ҳам бўлиши мумкин.

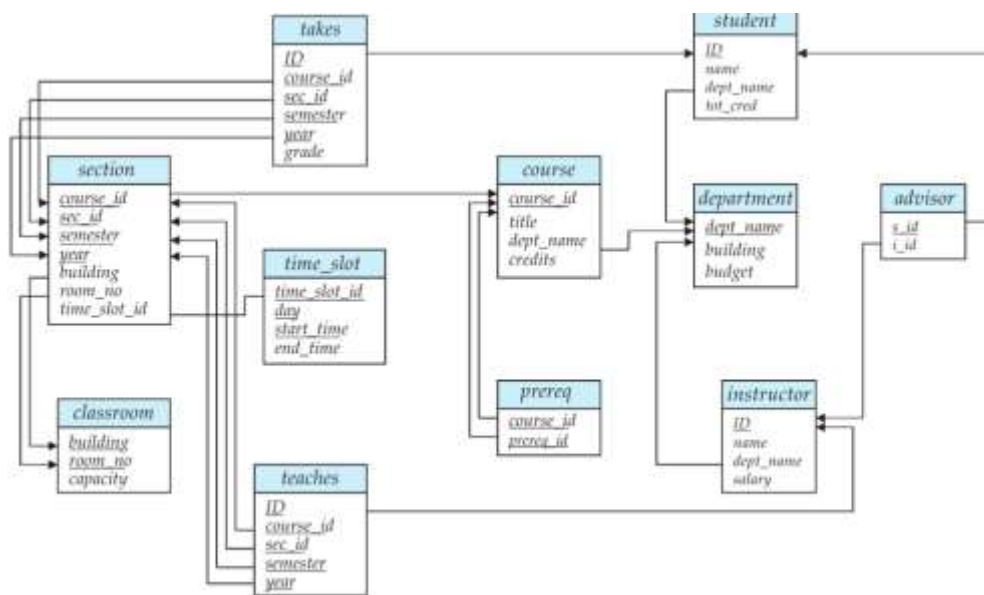
Модел схемаси¹¹

Маълумотлар базаси схемаси ,бирламчи ва иккиламчи калитлар билан биргаликда модел схемасида ифодаланиши мумкин. 1.10-расмда университет маълумотлар базаси модели схемаси келтирилган. Ҳар бир жадвал алоҳида тўртбурчакларда, уларнинг номлари шаклнинг юқори қисмида, жадвал

¹⁰ Silberschatz, Abraham. Database system concepts // 45 page

¹¹ Silberschatz, Abraham. Database system concepts // 46-47 pages

атрибутилари эса шакл ичида келтирилган. Иккиламчи калит кўрсаткич чизиқлар ёрдамида иккиламчи калитлардан бирламчи калитлар томон кўрсатилган.



1.11 – расм. Маълумотлар муносабати.

Мисол тариқасида ишлатаётган маълумотлар базаси университет ҳисобланади. 1.11 расмда берилган реляцион схема орқали кейинчалик ишлатадиган мисолларимизга қўлаймиз.

```

classroom(building, room_number, capacity)
department(dept_name, building, budget)
course(course_id, title, dept_name, credits)
instructor(ID, name, dept_name, salary)
section(course_id, sec_id, semester, year, building, room_number, time_slot_id)
teaches(ID, course_id, sec_id, semester, year)
student(ID, name, dept_name, tot_cred)
takes(ID, course_id, sec_id, semester, year, grade)
advisor(s_ID, i_ID)
time_slot(time_slot_id, day, start_time, end_time)
prereq(course_id, prereq_id)
    
```

1.12-расм. Маълумотлар

Реляцион сўровлар тили¹². Сўровлар тилида фойдаланувчи керакли маълумотларни олиш учун маълумотлар базасига сўров юборади. Бу тиллар одатда стандарт дастурлашт илларидадан бир поғона юқорида туради. Сўров тиллар икки гуруҳга процедуравий ва нопроцедуравийларга бўлинади. Процедуравий тилларда фойдаланувчи натижа олиш учун маълумотлар базасига маълум бир операциялар кетма-кетлигини амалга ошириши керак. Нопроцедуравий тилларда эса керакли маълумотларни олиш учун бу маълумотларга таъриф бериши керак бўлади.

¹² Silberschatz, Abraham. Database system concepts //47-48 pages

Реляционооперациялар. Кўпинча фойдаланиладиган операциялардан бири бу маълум бир шартга асосан ($salary > 85000$) маълум бир қортежни жадвалдан танлаш (инструктор). Янги ҳосил бўлган жадвал бу жадвал ости жадвалидир (инструктор).

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
12121	Wu	Finance	90000
22222	Einstein	Physics	95000
33456	Gold	Physics	87000
83821	Brandt	Comp. Sci.	92000

1.13-расм. Маълумотлар

Мисол учун юқоридаги жадваллардан “salary” 85000 катта бўлган қортежларни танлайдиган бўлсак, янги жадвал ҳосил бўлади. Яна бир бошқа кўп ишлатиладиган операциялардан бири бу жадвалдаги устунларни танлашдир. Бу операция натижаси танланган атрибутлар ташкил топган жадвал ҳосил бўлади. Мисол учун Инструктор жадвалидан “ID” ва “salary” атрибутини танлайдига бўлсак 1.14-расмдаги жадвал ҳосил бўлади.

<i>ID</i>	<i>salary</i>
10101	65000
12121	90000
15151	40000
22222	95000
32343	60000
33456	87000
45565	75000
58583	62000
76543	80000
76766	72000
83821	92000
98345	80000

1.14-расм. Маълумотлар

JOIN операцияси икки жадвални қўшишга ёрдам беради. Яъни жадвалдаги қортежларни битта қортежга қўшади. 1.15-расмда ва департамент жадвалларининг гўшилишдан ташкил топган жадвал келтирилган. Умуман олганда, JOIN операцияси иккита жадвал қортежидаги мос келган қийматлари бир хил бўлган атрибутларни қўшади.

Cartesian product операциясида эса JOIN операциясига нисбатан икки мос келмаган қийматларни ҳам қўшиш имконини беради. Юқорида айтилганидек, сўров натижалари асосида биз операцияларни амалга оширишимиз мумкин. Мисол учун, бизга инструктор жадвалидан 85000 дан

кўп маош оладиган ва “ID” ва “salary” атрибутларни чиқариш масаласи кўйилган бўлсин. Бу йерда биз биринчи навбатта 85000 дан кўп маош оладиганлар кортежини танлаб олишимиз керак. Бу сўров натижасидан келиб чиққан ҳолда “ID” ва “salary” атрибутларини танлаймиз. Ушбу сўров натижаси 1.15-расмда келтирилган.

ID	salary
12121	90000
22222	95000
33456	87000
83821	92000

1.15-расм. Маълумотлар

Баъзида сўров натижаси қайтариладиган кортежлардан ташкил топиши мумкин. Мисол учун инструктор жадвалидан “dept_name” атрибутини танласак, сўров натижаси “Comp. Sci.” қийматли кортежни уч маротаба такрорлаганини кўрамиз. Баъзи реляционтилллар математик кортеж атамасига яқинёндашган ҳолда қайтариладиган кортежлани бир маротаба ифодалайди. Баъзи реляцион тиллар сўров асосида олинган катта маълумотлар орасидан қайтарилган кортежларни ўчирмайди чунки бу кортежларни ўчириш учун кўп ресурс талаб етилишини ҳисобга олган ҳолда. Албатта вақт ўтиши билан маълумотлар базасидаги маълумотлар ўзгариши мумкин. Жадвал янги ёзувлар билан янгиланиши, мавжуд ёзувлар ўчирилиши ёки мавжуд ёзувлар ўзгартирилиши мумкин.

Реляцион алгебра¹³

Реляцион алгебра кўшиш, айириш ёки кўпайтириш каби одатий алгебраик операцияларни тўпламлар устида амалга ошириш имконини беради. Сонлар устида алгебраик операциялар бир ва ундан ортиқ сонларни қабул қилиб сон қийматини қайтаради, реляцион алгебра бир ёки ундан ортиқ тўпламларни қабул қилиб ягона тўпламни қайтаради. Реляцион алгебра чуқурроқ 6 бобда кўрсатилган. Қуйида бир неча операциялар келтириб ўтилган.

Символ(Номи)	Ишлатилиши
σ (selection)	$\sigma_{salary \geq 85000}$ (инструктор)
	Предикатни қаноатлантирувчи сатрларни қайтаради
Π	$\Pi_{ID, salary}$ (инструктор)

¹³ Silberschatz, Abraham. Database system concepts // 51 - pages

(Projection)	Жадвалдан берилган сўровга нисбатан ўзига хос жадвал чиқариш. Чиқувчи жадвалдан такрорланадиган кортежлар ўчирилади
\bowtie (Natural join)	инструктор \bowtie <i>department</i> Киритилган иккита жадвалдан бир-бирига мос келган атрибутларнинг қийматларидан ташкил топган жадвал
\times (Cartesian product)	инструктор \times <i>department</i> Киритилган иккита жадвалдан бир-бирига мос келган ва келмаган атрибутларнинг қийматларидан ташкил топган жадвал
\cup (Union)	Π_{name} (инструктор) \cup Π_{name} (студент) Иккита киритилган жадвални қўшиш

Назорат саволлари

1. Маълумотлар базасини бошқарув тизими билан ишлашда SQL тилининг роли хақида тушунча беринг?
2. SQL тилининг асосий вазифаси нималардан иборат?
3. SQL тили МВ да бошқариш вазифасини бажарадими?
4. SQL тили барча МВБТ ларида ишлатиладими?
5. SQL тили ва унинг яратилиш тарихи хақида тушунча беринг.
6. SQL тилининг стандартлари ва МВБТ версиялари диалектлари хақида тушунча беринг.
7. SQL ни ишлаш схемаси хақида тушунча беринг.
8. SQL тилининг қандай функционал имкониятлари мавжуд.
9. SQL тили бажарадиган функциялари.

Адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. см. 2011. ISBN 978-0-07-352332-3, chapter 1-2, page 57-112.
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)"(2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000).
5. www.natlib.uz.
6. www.ziyonet.uz.

2 - мавзу: SQL сўровлар тили (2 соат)

Режа:

- 2.1. SQL тили кириш.
- 2.2. Кўп сонли жадваллар устида сўровлар амалга ошириш.
- 2.3. Маълумотларни саралаш.
- 2.4. Бирлаштириш функциялари.

Таянч иборалар: *select, insert, delete, update, and, or, not, if, case, natural join, union амали, string амали, having сўзи.*

2.1. SQL тили кириш¹⁴

Ҳозирги кунда дунёда тижорий ёки экспериментал маълумотлар омбори сўров тиллари мавжуд. Бу бўлимда, шунингдек 4- ва 5 - бўлимда ҳам, энг кенг қўлланиладиган сўров тили – SQL кўриб чиқилади. Гарчи SQL тилини “сўровлар тили” деб айтилган бўлса ҳам, у маълумотлар омбори сўровидан кўра кўпроқ ишларни амалга ошириши мумкин. Мисол учун, у маълумотларнинг тузилишини аниқлайди, маълумотлар омборидаги маълумотларни ўзгартиради, хавфсизлик чеклашларини ўрнатади.

SQL сўровлар тили ҳақида қисқача маълумот. 1970 йилларнинг бошларида IBM компанияси, дастлабки номи Sequel бўлган, SQL нинг дастлабки кўринишини System R лойиҳасининг бир қисми сифатида ривожлантира бошлади. Шундан бери SQL тили ривожлантириб келинмоқда ва унинг номи SQL(Structured Query Language - Структураланган Сўровлар Тили)га ўзгарган. Кўпгина дастурий таъминотлар, ҳозирда, SQL тилини ишлатади. SQL, шубҳасиз, стандарт боғланишли маълумотлар омбори тили каби ташкил топган.

1986-йилда Америка Миллий Стандартлар Институти(American National Standards Institute - ANSI) ва Халқаро Стандартлаш Ташкилоти(International Organization for Standardization - ISO) SQL-86 деб номланувчи SQL стандартини чоп этди. Кейинчалик ANSI 1989-йилда SQL, SQL-89 стандартларини кенгайтириб чоп этди. Стандартнинг кейинги кўриниши SQL-92 бўлди. Шундан сўнг кетма-кет SQL:1999, SQL:2003, SQL:2006 стандартлари ва сўнгида эса SQL:2008 стандарти чиқарилди.

SQL тили бир қанча қисмлардан ташкил топган:

Маълумотлар-таърифлаш тили(Data-defeniton language – DDL). SQL DDL схемалар жадвалларини ўрнатишни, жадвалларни ўчиришни ва схема жадвалларини ўзгартиришни таъминлайди.

¹⁴ Silberschatz, Abraham. Database system concepts // 57 – pages

Маълумотларни қайта ишлаш тили(Data-manipulation language – DML). SQL DML маълумотлар омборидан маълумотларни чақиришни, маълумотлар омборига маълумот киритиш, уларни ўчириш ва маълумотларни ўзгартириш каби амалларни бажаради.

Умумийлик(integrity). SQL DDL калит майдонини ўрнатувчи буйруқларни ўз ичига олади. У калит майдонини мантиқий тузилишини бузувчи янгиланишларга йўл қўймайди.

Кўринишни аниқлаш(view definition). SQL DDL кўриниш(view)ларни аниқловчи буйруқларни ўз ичига олади.

Транзакцияларни бошқариш (Transaction control). SQL транзакцияларнинг бошланиши ва тугаши аниқловчи буйруқларни ўз ичига олади.

Ўзгармас SQL ва динамик SQL(Embedded SQL and dynamic SQL). Ўзгармас ва динамик SQL қандай қилиб SQL баёнотлари C, C++ ва Java каби умумий мақсадли дастурлаш тилларисиз ўрнатилиши мумкинлигини кўрсатади.

Авторизация(Authorization). SQL DDL жадваллар ва кўриниш(view)лардан эркин фойдаланиш ҳуқуқини белгиловчи буйруқларини ўз ичига олади.

Бу бўлимда биз SQLнинг асосий DML ва DDLларни текшириш хусусиятларини кўриб чиқамиз. Бунда SQL-92 SQL стандартидан кейин ишлатилиб келинаётган муҳим қисмлар ҳақида гапирилади.

4 - Бўлимда биз (а) турли хил ифодаларни бирлаштириш, (б)кўриниш(view)лар, (с) транзакциялар, (д) яхлитликларни чеклаш, (э) тизимларни ёзиш кабиларни кўриб чиқиш орқали SQL сўровлар тилини тўлароқ қамраб олишга ҳаркат қиламиз.

5-Бўлимда эса SQL тилининг SQL:1999 ва ундан кейинг версияларида тақдим қилинган (а) дастурлаш тилидан SQL га ўтишга рухсат берувчи механизмлар, (б) SQL функция ва процедуралари, (с) триггерлар, (д) рекурсив сўровлар ва (э) маълумотларни таҳлил қилиш тузилмаларини кўриб чиқиб, янаям кўпроқ хусусиятларини ўрганамиз.

SQL маълумотларни таърифлаш(SQL Data definition).

Маълумотлар омборидаги жадвалларни таърифлаш тизимда DDL (маълумотларни таърифлаш тили) воситаси орқали белгиланган бўлиши керак. SQL DDL фақат жадвалларни таърифлашигина эмас, яна қуйидагиларни ўз ичига олган ҳолда ҳар бир жадвал ҳақидаги маълумотни тавсифлашга ҳам имкон беради:

· Хар бир жадвалнинг схемасини.

- Қиймат турлари ҳар бир атрибут билан бирлаштирилишини.
- Калит майдони(умумийликни чеклаш)ни.
- Хар бир жадвал учун мавжуд бўлган кўрсаткич (индекс) ларни ўрнатишни.
- Хар бир жадвал учун хавфсизлик ва авторизация ахборотини.
- Хар бир жадвалнинг физик сақланиш тузилишини дискда сақлашни.

Асосий майдон типлари¹⁵. SQL стандартида турли хилдаги типлар ишлатилади:

- `char(n)`: фойдаланувчи томонидан ўрнатилган `n` ўзгармас узунликдаги белгили қатор. Унинг ўрнига тўлиқ кўриниши - `character` ишлатилиши ҳам мумкин.

- `varchar(n)`: фойдаланувчи томонидан ўрнатилган максимал узунлиги `n` та бўлган майдон узунликдаги белгили қатор. Бу майдоннинг тўлиқ кўриниши `character varying`.

- `int`: Бутун сон(ушбу тип узунлиги маша томонидан ўрнатилади). Тўлиқ кўриниши `integer`.

- `smallint`: кичик ўлчамли бутун сон.

- `numeric(p, d)`: ўзгармас нуқтали сонларни беришда ишлатилади. Бу ерда `p` - рақамлар(мусбат ишорали) ва `d` - ўнг томондан соннинг бутун қисмига берилувчи хоналар сони. Масалан `numeric(3, 1) 44.5`; . Бунда `numeric(3, 1)` га `444.5` ёки `0.32` каби сонларни бериб бўлмайди.

- `real, double precision`: кўзғалувчи нуқтали ва жуфт-ачегарали кўзғалувчан нуқтали сонлар. Чегаралари машина томонидан берилади.

- `float(n)`: кўғалувчан нуқтали жуда кичик `n` хонали сон .

Хар бир тип `null` деб номланувчи махсус қийматни ҳам ўз ичига олиши мумкин.

`char` маълумотлар типи ўзгармас узунликли қаторни беради. Мисол учун `char (10)` типли майдонни майдон А деб ҳисобласак. Агар биз бу майдонга “ `Avi` ” қаторини берсак, 10 та белги қилиш учун яна 7 та оралик(пробел) автоматик компьютер томонидан қўшилади. Таққослайдиган бўлсак, `varchar (10)` типли Б майдонга “ `Avi` ” қаторини сақламоқчи бўлсак, унда бу майдонга ҳеч қандай оралик(пробел) қўшилмасдан сақланади. Агар `char` типдаги бу иккала майдонни солиштирадиган бўлсак, уларнинг узунликлари бир хил бўлиши учун солиштиришдан олдин кўшимча пробеллар автоматик қўшилади ва кейин солиштирилади.

`Char` типи ва `varchar` типини солиштириш пайтида эса, уларнинг узунликларини тенглаштириш учун солиштиришдан олдин `varchar` типдаги

¹⁵ Silberschatz, Abraham. Database system concepts // 60 pages

майдонга қўшимча пробеллар қўшилади. Лекин шунга қарамай, солиштириш пайтида бу амалларнинг бажарилиши маълумотлар омбори тизимига боғлиқ. Натижа сифатида эса, А ва Б майдонларида бир хил “ Аvі ” қатори сақланса ҳам, солиштиришда А=В false қийматини қайтариши мумкин. Шунинг учун ҳам юқоридаги муаммони четлаб ўтиш учун чар типи ўрнига varchar типидан фойдаланган яхши.

Бундан ташқари SQL да яна Unicode белгиларидан фойдаланган ҳолда турли хил тилдаги маълумотларни сақловчи nvarchar типи ҳам ишлатилади. Аммо барибир ҳам, кўпгина маълумотлар омборлари Unicode(UTF -8 кўринишидаги) белгиларини varchar типидан сақлаш имкониятига эга.

Асосий схема таърифи¹⁶. SQLда жадвални яратишда create table буйруғидан фойдаланамиз. Қуйидаги буйруқ маълумотлар омборида ходимлар номли жадвални яратади:

```
create table ходимлар
    (исм varchar(20),
     хонаси varchar(15),
     маоши numeric(8,2),
     primary key (исм));
```

Юқорида яратилган жадвалда 3 та майдон мавжуд ва улар қуйидагилар:

исм - максимал белгилар сони 20 та бўлган белгили қатор майдони, хона - максимал белгилар сони 15 та бўлган белгили қатор майдони ва маоши - умумий хоналари сони 8 та, уларнинг 2 таси соннинг бутун қисмини ифодалаш учун кетган сон бўлган бутун қийматли сон майдони. Шу билан бирга create table буйруғи ичиқдаги умумийликни чеклаш буйруғи исм майдони ходимлар жадвалида асосий калит сўз(калит майдон) эканлигини кўрсатяпти.

Create table буйруғининг умумий кўриниши қуйидагича:

```
create table r
    (A1 D1,
     A2 D2
     ...
     An Dn,
     <kalit maydoni1>,
     ... ,
     < kalit maydoni n>);
```

Бу ерда r - жадвал номи, ҳар бир Ai – r жадвалнинг схемаси ичидаги майдон номи, Di – Ai майдоннинг типи.

¹⁶ Silberschatz, Abraham. Database system concepts // 60-61 pages

SQL турли хилдаги калит майдонлардан фойдаланиши мумкин. Қуйида уларнинг фақат бир нечтасини кўриб чиқамиз:

`primary key (Aj1, Aj2, ... , Ajm)`: `primary key Aj1, Aj2, ... , Ajm` лар жадвал учун асосий калит сўз эканлигини билдиради. `Primary` кей майдон қийматлари ҳеч қачон `null` бўлмаслигини ва такрорланмас бўлиши лозимлигини таъминлайди.

`foreign key(Ak1, Ak2, ... , Akn) references s`: Бу калит майдони жадвалдаги ҳар бир маълумотнинг `Ak1, Ak2, ... , Akn` майдонларининг қиймати `s` жадвалнинг баъзи маълумот майдонларининг қийматлари калит майдонларининг қийматига мос келиши кераклигини билдиради. Бу калит майдон жадвални бошқа жадваллар билан боғлашда хизмат қилади.

`not null`: ушбу калит майдони ўзи ўрнатилган майдон қийматининг `null` қиймат олиши мумкин эмаслигини билдиради.

SQL калит майдоннинг мантиқий тузилишини бузувчи ҳар қандай янгилашга йўл қўймайди. Мисол учун, жадвалнинг `primary-key` майдонига маълумот киритиш қисмига `null` қийматли маълумот киритилса ёки мавжуд маълумот `null` қийматлиси билан ўзгартирилса ва ёки `primary-key` майдоннинг маълумот киритиш қисмига киритилган қиймат шу жадвалнинг бошқа майдонига киритилган қиймат билан бир хил бўлса, SQL хатолик ҳақида маълумот чиқариб, янгилашни бекор қилади.

Маълумотлар омборида яратилган янги жадвал бўш бўлади. Биз `insert` буйруғи ёрдамида жадвалга маълумотлар киритишимиз мумкин. Масалан, биз жадвалга биология бўлимида ишловчи маоши 66 000 \$ бўлган `Smith` исмли ўқитувчини 10211 рақами билан киритмоқчимиз. Бунинг учун қуйидагича ёзамиз:

```
insert into instructor  
values(10211, 'Smith', 'Biology',66000);
```

Жадвалдаги маълумотлар қаторини `delete` буйруғи билан ўчириб ташлашимиз мумкин.

```
delete from student;
```

Бу буйруқ студент жадвалидан барча майдонларни ўчириб ташлайди. Ўчириш буйруғининг бошқа кўринишлари ўчирилган майдонларни аниқлаш имконини беради.

SQL маълумотлар омборидан жадвални олиб ташлаш учун `drop table` буйруғи ишлатилади. `drop table` буйруғи маълумотлар омборидан олиб ташланган жадвалнинг барча маълумотларини ўчириб ташлайди, аммо жадвалнинг ўзи маълумотлар омборида сақланиб қолади.

```
drop table r;
```

буйруғи

```
delete from r;
```

буйруғидан кўра кучлироқ.

Охирги буйруқ r жадвални сақлаб қолади, ammo ундаги барча майдонларни ўчириб ташлайди. Бундан олдинги буйруқ эса нафақат r жадвалнинг майдонларини балки r жадвалнинг схемасини ҳам ўчириб ташлайди.

Мавжуд жадвалга майдон қўшишда alter table буйруғидан фойдаланилади. Бунда жадвалга янги киритилган майдоннинг барча майдонлари null қиймати билан тўлдириб чиқилади. alter table буйруғининг умумий кўриниши қуйидагича:

```
alter table r add A D;
```

Бу ерда r - мавжуд жадвал номи, A - янги қўшилган майдон номи, D - майдоннинг тип.

Жадвалдаги майдонларни қуйидаги буйруқ ёрдамида олиб ташлаш мумкин:

```
alter table r drop A;
```

Бу ерда r - мавжуд жадвалнинг номи, ва A- жадвалдаги майдон номи. Гарчи кўпгина маълумотлар базаси тизимларида жадвални бутунлай олиб ташлаш имконияти бўлсада, улар майдонларни олиб ташлашни амалга оширмайди.

```
create table department17  
(dept_name      varchar(20),  
 building       varchar(15),  
 budget        numeric(12, 2),  
 primary key(dept_name) );
```

```
create table course  
( course_id     varchar(7),  
 title         varchar(50),  
 dept_name     varchar(20),  
 credits      numeric(2, 0),  
 primary key (course_id),  
 foreign key (dept_name) references department );
```

```
create table instructor  
( ID           varchar(5),
```

¹⁷ Silberschatz, Abraham. Database system concepts // 62-63 pages

```
name          varchar(20) not null,  
dept_name    varchar(20),  
salary       numeric(8, 2),  
primary key(ID),  
foreign key (dept_name) references department );
```

create table section

```
( course_id    varchar(8),  
  sec_id       varchar(8),  
  semester     varchar(6),  
  year         numeric(4, 0),  
  building     varchar(15),  
  room_number  varchar(7),  
  time_slot_id varchar(4),  
  primary key (course_id, sec_id, semester, year),  
  foreign key (course_id) references course );
```

create table teaches

```
( ID          varchar(5),  
  course_id   varchar(8),  
  sec_id      varcahr(8),  
  semester    varchar(6),  
  year        numeric(4, 0),  
  primary key (ID, course_id, sec_id, semester, year),  
  foreign key(course_id, sec_id, semester, year) references section,  
  foreign key(ID) references instructor );
```

Университет маълумотлар омборининг SQL маълумот таърифлаш қисми **SQL сўровларнинг асосий тузилиши.**¹⁸ SQL сўровларнинг асосий тузилиши учта сўздан иборат: select, from va where. Битта жадвалдаги сўровлар юқорида яратган Университет маълумотлар омборимиздан барча ўқитувчилар номини кўрмоқчимиз. Бунда қуйидаги буйруқни ёзамиз:

```
select * from instructor
```

Агар ўқитувчилар орасидан бирор name исмли ўқитувчини топмоқчи бўлсак, қуйидаги буйруқдан фойдаланамиз:

¹⁸ Silberschatz, Abraham. Database system concepts // 63-64- pages

```
select name from instructor
```

```
Name  
Нодир  
Карим  
Махмуд  
Хамдам  
Шохида
```

”select name from instructor” буйруғининг натижаси

Одатда, жадвал моделининг математик таърифида, жадвал - бу тўпلام деб ҳисобланади. Шундай экан, бир хил қаторлар жадвалда ҳеч қачон кўринмаган бўларди. Амалиётда эса, такрорланиб келувчи(бир хил қилиб киритилган) маълумотлар вақт сарфини бартараф қилади. Шунинг учун, SQL жадвалда, шу билан бирга SQL ифодасининг натижасида ҳам такрорланувчи маълумотларнинг ишлатилишига рухсат беради. Шундай экан, SQL instructor жадвалида кўринадиган ҳар бир қатор учун битта бўлим номи рўйхатини тузади.

Биз агар такрорланувчи маълумотларни олиб ташламоқчи бўлсак, такрорланувчи маълумотларни мажбурий бартараф қилиш ҳолатларида select калит сўзидан кейин distinct калит сўзини киритамиз ва юқорида ёзган буйруғимизни қуйидагига ўзгартирамиз:

```
select distinct dept_name from instructor;
```

Бу сўровдан кейин чиқувчи натижада ҳар бир бўлим номи фақат бир марта чиқади.

Жадвалдаги маълумотларни такрорланувчилари билан кўрмоқчи бўлсак all калит сўзидан фойдаланамиз:

```
select all dept_name from instructor;
```

Бунда такрорланувчи маълумотларни одатий ҳолда сақланмагунича all калит сўзидан фойдаланилмайди. Намуна сўровимизда натижадаги такрорланишларни бартараф қилишни сақлаш учун қачон зарур бўлса distinct калит сўзидан фойдаланамиз. select калит сўзи майдон қаторлари ёки константалар устида амал бажарувчи +, -, *, / ва / каби арифметик операторларни ҳам ишлатиши мумкин. Масалан, қуйидаги сўровни кўрайлик:

```
select ID, name, dept_name, salary * 1.1  
from instructor;
```

Бу сўров натижасида instructor жадвалидаги майдон қийматларидан фақат salary майдонининг қийматлари билан фарқ қилувчи қийматли жадвал

қийматлари чиқади. Бунда натижавий жадвал қийматидаги salaryнинг қийматлари эскисидан 10%га кўп бўлиб чиқади. Чунки юқоридаги сўровда salaryнинг қийматини 1.1 га кўпайтирдик.

where калит сўзи from калит сўзи томонидан чақирилувчи жадвалнинг берилган қаторини натижа сифатида чиқариш имконини беради. Фараз қилайлик “Информатика Бўлимида ишловчи, маоши 700000 дан кўп бўлган барча ўқитувчиларни чиқарсин”. Бу сўров буйруғи қуйидагича:

```
select name
from instructor
where dept_name = `Informatika` and salary>700000;
```

SQL да where калит сўзи билан биргаликда мантикий and, or ва not боғловчиларини ҳам ишлатиш мумкин. Мантикий боғловчиларни қуйидагича ҳам ифодалаш мумкин:

<, <=, > >= = ва <>.

2.2. Кўп сонли жадваллар устида сўровлар амалга ошириш

Шу пайтгача биз битта жадвал устида амалга оширилган сўровларни кўриб чиқдик. Сўровлар кўпинча бир нечта жадваллар маълумотлари билан ишлашда керак бўлади. Фараз қилайлик “Бўлим номлари ва бино номлари бўйича барча ўқитувчиларнинг исмларини аввалги ҳолатига қайтарайлик”. instructor жадвали схемасига қараб, биз бўлим номини dept_name майдонидан олишимизни биламиз, лекин Бўлим биноси номи департамент жадвалининг буилдинг майдонида берилган. Сўровга жавоб бериш учун instructor жадвалидаги ҳар бир қатор dept_name қиймати instructor қаторининг dept_name қийматига солиштириладиган департамент жадвалининг қатори билан солиштирилиши керак.

Юқоридаги сўровга SQLда жавоб бериш учун, биз from калит сўзи билан мурожаат қилиниши керак бўлган жадваллар рўйхатини тузамиз ва where калит сўзи билан солиштирамиз:

```
select name, instructor.dept_name, building
from instructor, department
where instructor.dept_name = department.dept_name;
```

Хаммага маълумки, SQL сўровлари уч хил турдаги - select, from ва where каби калит сўздан иборат. Ҳар бир калит сўзнинг қуйидагича вазифаси мавжуд:

· Select калит сўзи сўровнинг натижаси сифатида майдоннинг маълумотлари рўйхатини тузишда фойдаланилади.

· From калит сўзи жадвал рўйхатидан сўров бўйича сўралган жадвал ёки

жадвалларга мурожаат қилади.

· Where калит сўзи фром калит сўзи томонидан чақирилган жадваллар майдонларининг кераклисини ажратиб олади.

SQL сўровнинг умумий кўриниши:

```
select A1 , A2 ,..., An
from r1 , r2 ,...,rm
where P;
```

Бу ерда A_i лар майдонлар, r_i - жадваллар ва P- ажратиб олинган қийматлар.

Natural join

```
select name, course_id
from instructor, teaches
where instructor. ID = teaches. ID and instructor.dept name = 'biology';
```

Бу мисолда instructor ва teaches жадвалларидан олинувчи маълумотларни чиқарувчи сўров instructor нинг ID сини teaches нинг ID си билан солиштиряпти. Булар иккита жадвалдаги бир хил номли майдонлардир. Аслида олганда бу умумий ҳолатдир. Бунда from калит сўзидаги ҳолатларни солиштирилиши номларининг бир хиллигини солиштирувчи барча майдонларни кўп маротаба сўрайди(ишлатади).

Бу умумий ҳолатда SQL дастурчининг ишини осонлаштириш учун SQL natural join деб номланувчи амални ишлатади. Аслини олганда SQL икки ёки ундан кўп жадвалларнинг маълумотларини биргаликда бирлаштирувчи бошқа бир қанча усулларни қўллайдди.

Natural join амали иккита жадвалнинг устида амал бажаради ва натижа сифатида битта жадвал чиқаради.

“Университетда бир хил курсга дарс ўтувчи ўқитувчиларнинг исмини ва улар дарс берувчи барча курсларнинг ID сини топиш” масаласининг сўров буйруғини қуйидагича ёзилган:

```
select name, course_id
from instructor, teaches
where instructor. ID = teaches. ID;
```

Бу масала сўрови SQLнинг ўз-ўзидан бирлашиш амали ёрдамида анча қисқартирилиши мумкин:

```
select name, course id
from instructor natural join teaches;
```

SQL сўровда from калит сўзи натурал joinдан фойдаланиб, кўп сондаги жадвалларни бирлаштирилиши мумкин. Унинг кўриниши қуйидагича бўлади:

```
select A1 , A2 ,..., An
from r1 natural join r2 natural join . . . natural join rm
```


where P;

Умумий ҳолатда, from калит сўзи қуйидаги кўриниш билан бир хил бўлиши мумкин:

```
from E1 , E2 , . . . , En
```

Бу ерда E_i битта жадвал ёки natural join ларни ўз ичига олувчи ифода бўлиши мумкин

Natural join га мисол:

```
select name, title
```

```
from instructor natural join teaches natural join course;
```

майдонларни тенглаштириш хатолигини четлаб ўтиш пайтида natural join амалини яхшилаш мақсадида SQL қайси устунлар тенглаштирилиши кераклигини аниқлаш имконини берувчи natural join тузилишининг кўриниши таъминлайди. Бу тузилиш қуйидаги сўров орқали тасвирланган:

```
select name, title
```

```
from (instructor natural join teaches) join course using (course id);
```

join ... using амали аниқланиши керак бўлган майдон номларининг рўйхатини талаб қилади. Иккала киришлар ҳам аниқ номли майдонларга эга бўлиши керак. r1 join r2 using(A1, A2) амали r1 natural join r2

амали билан бир хил.

SQLда бир қанча қўшимча муҳим амаллар ишлатилади.

Қайта номлаш (Rename) амали.

```
select name, course_id
```

```
from instructor, teaches
```

```
where instructor. ID = teaches. ID;
```

Бу сўровнинг натижаси сифатида name ва course_id жадваллари чиқади. Лекин баъзи ҳолатларда майдонларни ёзиш пайтида майдон номларини ўзгартиришга тўғри келади. Бунда биз керакли майдон номини сўров буйруғи кодларида ўзгартирамиз. Бунда қуйидаги коддан фойдаланилади:

```
eski_nom as yangi-nom
```

Юқорида ёзган кодимиздаги name майдон номини instructor_name га ўзгартирмоқчи бўлсак қуйидагича сўров ёзилади:

```
select name as instructor name, course id
```

```
from instructor, teaches
```

```
where instructor. ID = teaches. ID
```

String (қатор) амаллари

Жадвал майдони номлари ёки бошқа қаторли қийматлар билан ишлашга тўғри келганда баъзан турли хил муаммоларга учраймиз. Мисол учун

харфларнинг катта кичиклиги бўйича юз берувчи муаммоларни олиш мумкин. Айрим маълумотлар омбори тизимлари қаторларни солиштиришда катта ва кичик ҳарфларни фарқламайди. Бу муаммоларни ҳал қилишда қатор амали функцияларидан фойдаланилади.

SQL да қаторлар устида қуйидагича амаллар мавжуд:

- Боғлаш. Бунда “||” белгисидан фойдаланилади

- Қаторостиларни чиқариб олиш.

- Қатор узунлигини топиш.

- Қаторни юқори регстрга ўтказиш. upper(s) функциясидан фойдаланилади. Бу ерда s ўзгартирилувчи қатор.

- Қаторни қуйи регстрга ўтказиш. lower(s) функциясидан фойдаланилади.

- Қатордаги оралик(пробел)ларни олиб ташлаш. trim(s) функциясидан фойдаланилади.

Турли хилдаги маълумотлар омбори тизимларида қатор функциялари турлича бўлади. Бунинг ўзининг маълумотлар омбори тизмининг маълумотномасидан кўриб олишингиз мумкин.

Берилган қиймат билан солиштиришда лике операторидан фойдаланилади. Берилувчи қийматларни махсус белгилар орқали тасвирлаймиз:

- Фойиз(%): % белгиси бирор қатор қисмини солиштиришда фойдаланилади.

- Ост чизик(_): _ белгиси бирор белгини солиштиришда ишлатилади.

Барча майдонларни танлаш

Сўровда барча майдонларни танлашда “*” белгисидан фойдаланилади.

```
select instructor.*
```

```
from instructor, teaches
```

```
where instructor. ID = teaches. ID
```

Бу ерда instructor жадвалининг барча майдонлари танланди.

2.3. Маълумотларни саралаш

Жадвалдаги маълумотлар устида иш олиб борилганда уларни тартибланишга тўғри келиб қолади.

Бу ҳолатда SQL да “order by” амалидан фойдаланилади. Тартибланиш ўсиш ёки камайиш тартибида бўлиши мумкин. Камайиш тартибида сараланишда desc амалидан, ўсиш тартибида сараланишда эса asc амалидан фойдаланилади.

Мисол.

```
select *
```

```
from instructor
order by salary desc, name asc;
```

Where калит сўзи ёрдамида маълумотларни саралаш. SQL да where калит сўзи ёрдамида маълумотларни саралашда between таққословчи калит сўзидан фойдаланилади. Мисол учун маоши 900000 ва 1000000 бўлган ўқитувчиларни сараласак:

```
select name
from instructor
where salary between 90000 and 100000;
ёки
select name
from instructor
where salary <= 100000 and salary >= 90000;
```

Тўплам амаллари

SQL да union, intersect va except амаллари жадваллар устида математикадаги тўплам амаллари бўлган - \cup , \cap ва - амаллари бажарган амалларни бажаради.

Union амали. Иккита жадвалдаги қидирилаётган барча маълумотларни чиқариш учун ишлатилади.

2009 -йилда қуйи семестерда ёки 2010 - йил баҳорда ўқитилган барча курслар тўпламини чиқариш:

```
(select course_id
from section
where semester = 'Fall' and year= 2009)
union
(select course_id
from section
where semester = 'Spring' and year= 2010);
```

Intersect амали

Қидирилаётган маълумотнинг иккита жадвалда ҳам бор бўлганини чиқариш учун ишлатилади.

2009 -йилда қуйи семестерда ва 2010 - йил баҳорда ўқитилган барча курслар тўпламини чиқариш:

```
(select course_id
from section
where semester = 'Fall' and year= 2009)
intersect
(select course_id
```

from section

where semester = 'Spring' and year= 2010);

Intersect амали

Қидирилаётган маълумотнинг иккита жадвалда ҳам бор бўлганини чиқариш учун ишлатилади.

2009 -йилда қуйи семестерда ва 2010 - йил баҳорда ўқитилган барча курслар тўпламини чиқариш:

(select course_id

from section

where semester = 'Fall' and year= 2009)

except

(select course_id

from section

where semester = 'Spring' and year= 2010);

Null қиймати

Null қийматлар ўз ичига арифметик, солиштириш ва тўплам амалларини олувчи боғланган амалларда алоҳида муаммоларда қатнашади.

Агар киритилган қиймат null бўлса, арифметик ифоданинг ҳам натижаси null қиймат бўлади. Масалан, агар сўровда $r.A + 5$ деган ифода мавжуд бўлса ва $r.A$ нинг бирор қатори null қийматида бўлса, шу қатор ифодасининг ҳам қиймати null қийматли бўлади.

Таққослаш ифодаларида null қиймати кўпроқ муаммо келтириб чиқаради. Масалан, “ $1 < null$ ” деб олинса, бунда муаммо пайдо бўлади: null бирдан каттами ёки кичик. Чунки null нинг қиймати номаълум. Мантиқий амалларда эса “ $not(1 < null)$ ” амалининг қиймати true чиқади.

Мисол:

select name

from instructor

where salary is null;

Бу ерда сўров натижасида маоши null қийматида бўлмаган ўқитувчиларнинг номлари рўйхати чиқади.

2.4. Бирлаштириш функциялари

Бирлаштириш функциялари - бу киритилган қийматларнинг тўпламини олиб, битта қиймат қайтарувчи функциялардир. SQL да беш турдаги бирлаштириш функциялари мавжуд:

- Ўртача қийматни чиқариш: avg
- Минимумни чиқариш: min

- Максимумни чиқариш: max
- Умумий ҳисобни чиқариш: sum
- Сонини чиқариш: count

Булардан сум ва avg сонли фақат қийматлар устида амаллар бажаради. Қолганлари эса сон бўлмаган маълумот қийматлари устида ҳам амаллар бажаради.

Асосий бирлаштиришлар

Computer Science бўймидаги ўқитувчилар маошининг ўртача ҳисобини топиш:

```
select avg (salary)
from instructor
where dept_name= 'Comp. Sci.';
```

Гуруҳли бирлаштиришлар

Биз бирлаштириш амалларини нафақат битта қатор, балки қаторлар тўплами устида ҳам амалга оширишимиз мумкин. Бунда биз group by калит сўзидан фойдаланамиз.

Мисол. Хар бир Бўлимнинг маошлари ўртача қийматларини топинг.

```
select dept_name, avg (salary) as avg_salary
from instructor
group by dept_name;
```

Хавинг калит сўзи

Гуруҳли функцияларни, шарт қўйишда ишлатганда «хавинг» калит сўзидан фойдаланилади, чунки «where» калит сўзи гуруҳли функциялар билан ишлай олмайди. Агар, «хавинг» ўрнига «where» ишлагангиз, хатолик беради. Бу операторни қисм сўровлар тузишда ҳам ишлатиш мумкин.

Назорат саволлари

1. Университет маълумотлар базасини яратиш умумий структураси?
2. Университет маълумотлар базасини яратишда қандай SQL сўровлари ишлатилади?
3. SQL да Кафедра сўровларини яратинг?
4. SQL да малака ошириш маълумотлар базасини яратинг сўровларини яратинг?
5. Икки жадвалли сўровни қандай яратиш мумкин.
6. Маълумотларни янгилаш учун қандай оператор фойдаланилади.
7. Маълумотларни қайта ишлашда энг мураккаб операторнинг вазифаси нима?

Фойдаланилган адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. cm. 2011. ISBN 978-0-07-352332-3, chapter 1-2, page 57-112.
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)"(2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000).
5. <http://www.colapublib.org/periodicals/>
6. <http://houstonlibrary.org/research/databases>

3 – мавзу: Маълумотлар базасини лойиҳалаштириш (2 соат)

Режа:

- 3.1. Кириш.
- 3.2. Лойиҳалаш жараёни.
- 3.3. Лойиҳалашда қарор қабул қилиш.
- 3.4. Моҳият муносабат модели.
- 3.5. Реляцион маълумотлар базаси дизайни.

Таянч иборалар: *лойиҳалаш, моделлаштириш, реляцион моделлар дизайни, маълумотлар базаси, жадвал, муносабат, моҳият.*

3.1. Кириш

Мавзудаги фикрга боғлиқ ҳолда берилган маълумот баъзаси схемасини ва ундаги сўровларни қай ҳолатда амалга оширишни кўрсатиш мутахасисдан таълаб эътилади. Дастлаб Маълумотлар Базаси схемасини қандай кўринишда бўлиши ҳақида сўз юритамиз. Ушбу бўлимда моҳият -алоқа моделига кўпроқ эътибор қаратамиз. Моҳият алоқа модели нималардан ташкил топган ва улар билан қандай боғланганлигини кўриб чиқамиз. Натижада, маълумотлар база (МВ)си дизайни боғланган ҳолдаги МВ си кўринишига келади ва буйруқлар билан бирлаштирилади. Ушбу бўлимда биз моҳият-алоқа модели лойиҳасини қандай қилиб схемадаги алоқаларини(боғлиқликларини) ўрнатиш ва амаллар қандай ишлашини кўрсатиб ўтамиз.

3.2. Лойиҳалаш жараёни¹⁹

МВ қатнашган исталган исталган дастурий воситани яратиш бу бир қанча мураккаб жараёндинир. Бу жараён МВси схемасининг дизайн қисми, дастур кўринишини ва кўрсаткичларини янгилаш, киритиш каби жараёнларда ўзининг муҳим ўрнини тутди. Юқорида таъкидлаганимиздек, ушбу бўлимда МВ чизмасининг кўринишига кўпроқ аҳамият берамиз.

Лойиҳа Босқичи

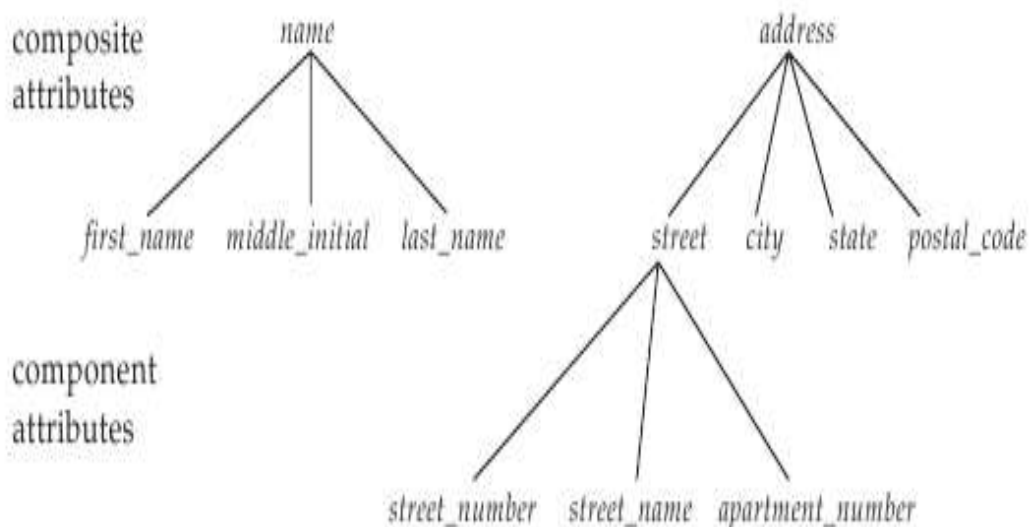
Кичик дастурлар учун яратилиши керак бўлган алоқаларда тўғридан-тўғри мурожаатни ўрнатиш дастур учун керакли бўлган атрибутлар ва буйруқларни тушинарли қилиб яратиш МВ дизайнерлари амалга ошириш мумкин бўлган ҳолат ҳисобланади. Аммо, шундай бир кўринишдаги жараён мавжудки у реал-ҳаётий дастурда мураккаблик туғдиради. Уларнинг

¹⁹ Silberschatz, Abraham. Database system concepts // 266-268 - бетлар

мураккаблик даражаси юқори бўлади. МВ дизайнерлари керак бўлганда фойдаланувчига дастурни тушунарли ҳолатга келтиришга ўз таъсирларини кўрсатишлари керак. Шунингдек, дастур учун керакли маълумотларни киритиш усулларини қуйи даражада тушунарли ҳолатда яратиш лозим. Юқори даражадаги маълумотлар моделлари МВ дизайнерларига махсус концептуал кўринишлардагина тизимли кўринишларда, махсус мутахасислар учун яратилинади.

Дастлабки босқич МВ лойиҳасининг келгусидаги МВ си фойдаланувчиларининг тўлақонли маълумотларини ифодалаб чиқишдан иборатдир. МВ лойиҳаси ушбу босқичда фойдаланувчилар талабини ўзига хослигини ифодалайди. Яъни фойдаланувчи талабларини диаграммалар асосидаги техникасида кўрсатади. Айнан шу босқичда фойдаланувчилар талаблари ёзма ҳолатда ифодаланишига чек қўйилади.

Кейинги қадамда, мутахасислар маълумотлар модулини танлайдилар ва танланган модел ғояси асосида фойдаланувчилар талабларини МВ нинг қоралама (синалмаган) чизмасига кўчиришади. Чизма(схема) ушбу “синалмаган” кўриниш босқичида танланган модел ғоясида барча тасаввурларни жамлаб шу босқичда керакли амалларни бажаради. Моҳият алоқа модели худди шу модел асосидаги чизмаларда намоён бўлади. Моҳият алоқа модели мобайнида концептуал схема МВда боғлиқликлар, моҳиятнинг атрибутлари, моҳиятдаги буйруқлар ва алоқалар намоён бўлади. Ўз ўрнида концептуал(қоралама) чизма босқичидаги натижа диаграммаси моҳиятни яратиш алоқаларини чизмада намоён этиб беради.



3.1 – расм. Лойиҳалаш

Тўлалигича қайта ишланган концептуал схема ишланаётган объектнинг функционал талабларини кўрсатади. Функционал талабларнинг хусусиятида МВ да кўринадиган амаллар турлари фойдаланувчи томонидан

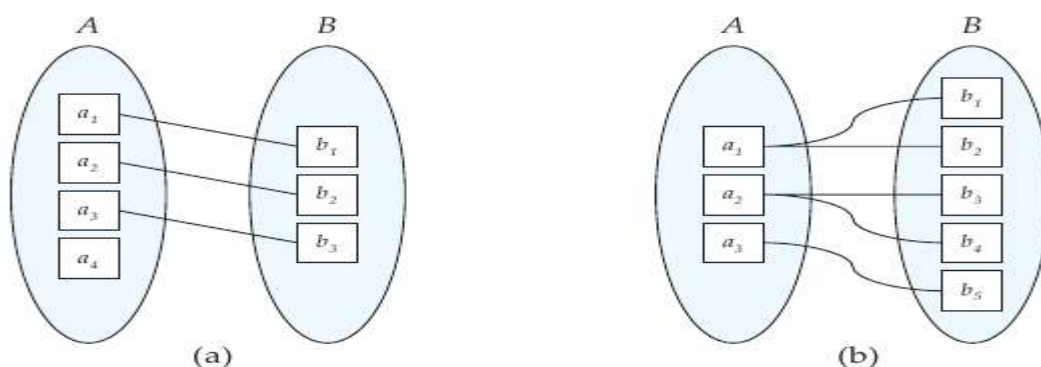
тасвирланиши кутилади.

Маълумотларни янгилаш, ўзгартириш, кидириш, олдинги ҳолатига қайтариш ва маълумотни ўчириш шулар жумласидаги амалларга мисол бўлади. Концептуал лойиҳалаш босқичида мутахасис схема учраши мумкин бўлган функционал талабларни олдиндан кўра билиши мумкин.

Мавхум маълумот моделидан МВ нинг икки сўнги лойиҳалаш қадамларга кўчириш билан жараён давом этади.

- Мантикий лойиҳалаш босқичида, мутахасисдан фойдаланиладиган МВ тизимининг маълумотлар модели амалга оширилишида юқори даражадаги концептуал схемани тақдим этиши кутилади. Маълумотлар моделининг амалга ошиши бевосита унинг танланган маълумотлар моделига боғлиқдир. Ушбу қадамда, боғлиқлик схемасига моҳият-алоқа моделини тасвирловчи чизма кўрсатилади.

Тўлалигича қайта ишланган концептуал схема ишланаётган объектнинг функционал талабларини кўрсатади. Функционал талабларнинг хусусиятида МВ да кўринадиган амаллар турлари фойдаланувчи томонидан тасвирланиши кутилади.



3.2 – расм. Маълумотларни лойиҳалаш

- Хуллас, лойиҳа мутахасиси кейинг физик лойиҳа босқичида тизимнинг махсус МВ схемаси натижасидан фойдаланилади.

- МВ нинг физик схемаси дастур яратилгандан сўнг осонгина ўзгартирилиши мумкин. Бироқ, мантикий схемадаги ўзгартирилиш ўзида бир қанча қийинчиликларни олиб келади чунки дастур кодидаги сўровлар ва янгиланишлар буйруқлар сони бунга таъсир кўрсатади. Шу сабабдан ҳам МВ ни лойиҳалаш қадамлари босқичма-босқич амалга оширилиши муҳим саналади.

- Хуллас, лойиҳа мутахасиси кейинг физик лойиҳа босқичида тизимнинг махсус МВ схемаси натижасидан фойдаланилади.

- МВ нинг физик схемаси дастур яратилгандан сўнг осонгина

Ўзгартирилиши мумкин. Бироқ, мантикий схемадаги ўзгартирилиш ўзида бир қанча қийинчиликларни олиб келади чунки дастур кодидаги сўровлар ва янгилашлар буйруқлар сони бунга таъсир кўрсатади. Шу сабабдан ҳам МВ ни лойиҳалаш қадамлари босқичма-босқич амалга оширилиши муҳим саналади.

3.3. Лойиҳалашда қарор қабул қилиш

МВ лойиҳалашнинг асосий қисми бу- инсонлар, жойлар ва маҳсулот номларининг турларини қандай кўрсатиб бериш масаласи ҳисобланади. Биз моҳият модели мобайнида бизга маълум бўлган номларни боғлашда фойдаланамиз. Университет МВси мисолида моҳият модели раҳбар ўқитувчилар, талабалар, бинолар, ўқув курслари ва ўқув курсларига таклифлар ушбу модел қатнашчилари бўлади.

Турли хилдаги моҳиятлар бир-бири билан боғланади. Буларнинг барчаси МВ лойиҳасида акс этиши шарт бўлган омилдир. Масалан, таклиф қилинаётган курс мисолида шу курсни оқиш учун талаба мурожаат қилади ва айни бир ҳолда ўқитувчи ҳам унга шу курсни ўқитиш учун мурожаат этади. Ўқиш ва ўқитиш битта объектга мурожаати моҳиятлар орасидаги алоқага мисол бўлади.

МВ чизмаси лойиҳасида 2 турда юзага келиши мумкин бўлган муаммодан қочиш керак:

1- Ортиқча такрорланиш. Ёмон лойиҳада маълумотлар такрорланиб қолади. Масалан, Курс мисолида таклиф қилинаётган ҳар бир курснинг тартиб номери билан номини сақласак, унг мурожаат вақтида курс номи ортиқча ҳолатда пайдо бўлаверади. Бу ҳолат фойдаланувчига салбий таъсир этади.

2- Етишмовчилик. Ёмон лойиҳаланган МВ сида қийинчиликлар аниқ, яққол кўрина бошлайди. Бунда танлабган модел хато бўлиб чиқади. Масалан, юқоридаги мисолда тасаввур қилинг, ўқув курсининг ўзи учун бизда мос моҳиятлар бор бўлсину лекин таклиф қилинаётган курсларда улар мавжуд бўлмасин. Тенг ҳолатда боғлиқлик жараёнида таклиф қилинган ҳар бир курс учун маълумот такрорланган жойда битта алоқа ўрнатилган. Бу ҳолат кейинчалик таклиф қилинаётган янги курсларни чоп этишни йўққа чиқаради.

3.4. Моҳият муносабат модели²⁰

Моҳият алоқаси маълумотлар модели - МВ нинг мантикий структурасини таърифловчи мукамал схеманинг хусусиятларини ўзида жамлаган МВси моделидир. Моҳият алоқа модели концептуал схемаси устида қурилган реал-ҳаётий босқичларни ўзида жамлаган фойдали модел ҳисобланади.

²⁰ 1. Silberschatz, Abraham. Database system concepts // pages- 292 - 293

76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

instructor

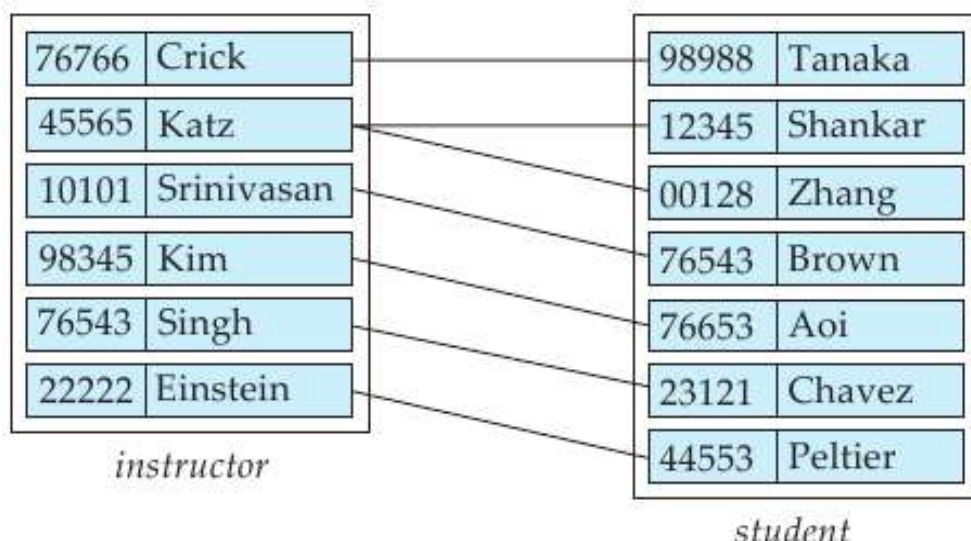
98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

3.3 – расм. Маълумотлар

Моҳиятни ўрнатиш. Моҳият -бу объект ёки “нарса” ҳақиқий ҳаётда мавжуд ва бошқаларидан бўлган фарқланадиган бўглиқликдир. Масалан, ҳар бир инсон университет мисолида моҳият бўла олади. Моҳият- ўзининг хусусиятларига эга бўлиб, хусусиятларининг қийматини ўрнатиш учун моҳиятга тартиб рақами берилади. Инсонлар мисолида person_id хусусияти шу инсонга берилган ID билан аниқланади. Шу нуқтаи назарда 677-89-9011 университетдаги қандайдир инсонни аниқлайди.

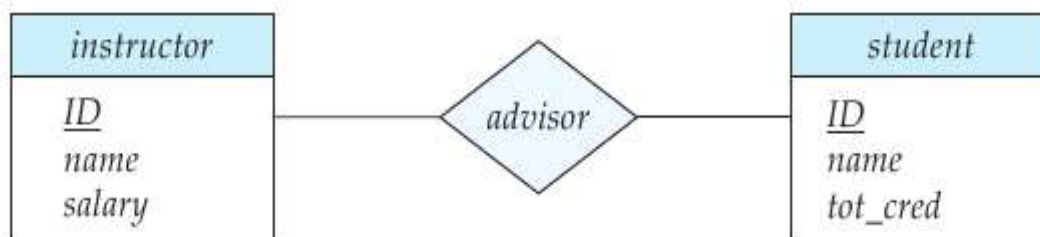
Моҳиятни ўрнатиш бир хил атрибут хусусиятдаги ва турларни тартиблаш имконини беради. Моҳият атрибутларининг ўрнатилиши орқали намоуйиш қилинади. Атрибутлар- моҳиятнинг ҳар бир аъзоси томонидан эргашиб унинг хусусиятини тасвирлаб келади. Реал ҳаётда қўшимча атрибутлар қуйидагича бўлиши мумкин: кўча номлари, бино рақамлари, почта кодлари, туманлар. Misol учун course_id, dept_name, title_id.



3.4 – расм. Маълумотлар

Юқоридаги чизмада ўқитувчилар ва талабалар орасидаги ўрнатилган боғлиқлик муносабатини кўриш мумкин. Боғлиқликларгаги моҳиятнинг кўрсатаётган ушбу амали моҳиятнинг модел роли деб юритилади.

Асосий структура



3.5 – расм. Маълумотлар

Моҳиятни ўрнатиш жараёнида тўғри-тўртбурчакли фигура чизмаси икки қисмга ажратилинади. Биринчи қисмдаги кўк рангли ном ўзида моҳиятлар ўрнатилган номларни кўрсатади. Иккинчи оқ рангдаги қисмда эса моҳиятга берилган атрибут номлари келтирилади.²¹

Ромб кўринишидаги белги ўрнатилган алоқани ифодалайди.

Яхлит ҳолатдаги тўртбурчак алоқалар ўрнатилган атрибутларни ўз ичига олади. Бошланғич калитли атрибутлар тагига чизик билан ажратилади.

Фойдаланилган тўғри чизиклар боғлиқликларни ўрнатади.

Йўналиши кўрсатилган чизиклар боғлиқлик ўрнатиладиган атрибутларга йўналтирилади. Параллел чизиклар боғлиқликларни ўрнатишда моҳиятнинг умумий қатнашишини билдиради. Иккиланган ромб тўлиқ таъминланмаган моҳиятларда алоқа ўрнатилганликни тасвирлайди.

²¹ 1. Silberschatz, Abraham. Database system concepts // pages-302-306



(a)



(b)



(c)



(a)



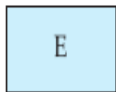
(b)



(c)

3.6 – расм. Маълумотлар

UML - Unified Modeling Language. Маълумотларни моделлаштириш учун бир қанча хилдаги шартли белгиланишлардан фойдаланилади. Моҳият алоқа модели ва UML синф диаграммаларидан кенг кўламда фойдаланилади. Кўпгина адабиёт манбаларида бу белгилар бир хил бўлавермайди.



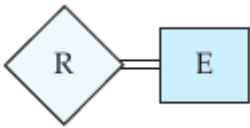
Моҳиятни ўрнатиш ;



Боғлиқликни ўрнатиш;



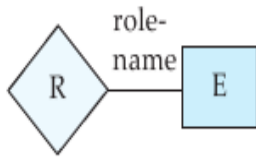
Кучсиз моҳиятдаги алоқа;



моҳият умумий қатнашуви



Бирга-бир боғланиш



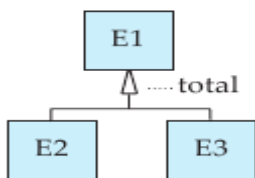
кўрсаткичли амал



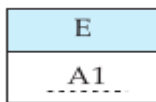
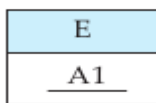
кўпга-кўп боғланиш



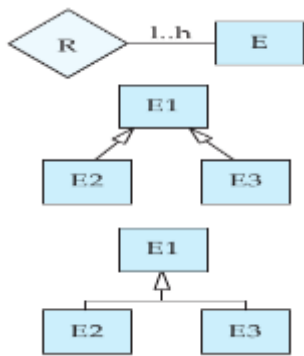
Кўпга бир боғланиш



Умумлаштириш

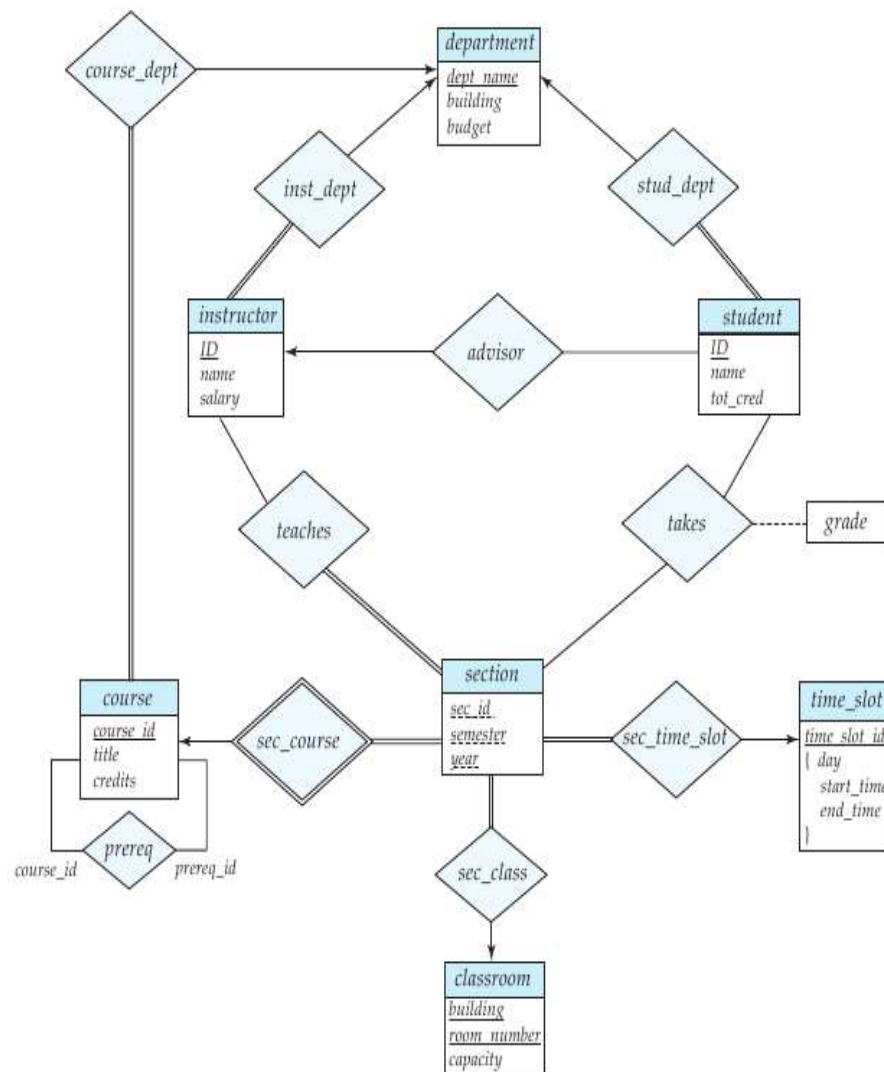


Бошланғич калит ва моҳият атрибути



Амаллардаги чекловлар;

Махсуслаштириш ва умумлаштириш ҳолда Университет жараёнидаги Моҳият-Алоқа Диаграммаси



3.7-расм. Алоқа Диаграммаси

Юқоридаги чизма университет мисолида амалга ошувчи жараёнлар Моҳият-Алоқа диаграммасида ёритилган. Университет МВсида шундай турдаги чеклов мавжудки унга кўра ҳар бир ўқитувчи ўзига бириктирилган бинода бўлиши керак. Натижада, тасвирда кўриниб турган ўқитувчи (instructor) ва бино (inst_dept) орасига умумий қатнашувини билдирувчи

параллел чизиқлардан фойдаланилган. Қўшимча қилиб айтганда, институтдан департаментга йўналган кўрсаткичли тўғри чизиқ ҳар бир ўқитувчи кўпи билан битта бинога тегишли эканлигини ифодалайди.

Шунга ўхшаб, мос равишда ўқув курси (course) ва талаба (студент) моҳиятлари орасидаги муносабатларда умумий қатнашувни ифодалашда параллел чизиқлар қўлланилган. Уларнинг умумлашишини департамент да кузатиш мумкин.

Чизмада яна ҳар бир талаба кўпи билан битта маслаҳатчи олиши мумкинлиги кўрсатилган ва section бўлимида ўрнатилган моҳият кучсиз бўлиб, sec_ид, семестер ва ear атрибутлари билан кўрсатилган.

UML тили (Стандарт моделлаш тили)

Моҳият алоқаси диаграммаси дастурий маҳсулот тизимининг маълумот компонентларини ўзининг диаграммасида яққол кўрсатади. Маълумотлар намоиши тизимнинг фақатгина бир қисми ҳисобланади. Бошқа компоненталар тизим билан биргаликда ўзаро муносабатларни ҳам кўрсатади. Буларга тизимнинг функционал моделлари, уларни синфлашлар ва улар орасидаги амаллар мисол бўлади. UML (Стандарт Моделлас Тили) Объектларни бошқариш гуруҳи (Object Манагемент Групп) ҳомийлигида дастурий тизимнинг компоненталарини махсус яратиш мақсадида стандартлаштирилган.

UML нинг бази қисмлари:

1-Синфлаш диаграммаси (Class диаграм).

Синфлаш диаграммаси Моҳият-Алоқа моделига ўхшатиб яратилинади.

2-Фойдаланувчи диаграммаси (Use Case Диаграм)

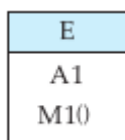
Фойдаланувчи диаграммаси унга дастурдан қандай қилиб фойдаланиш кўрсатмаларида ёрдам беради. Масалан, ўз ҳисобидан пул ўтказиш ёки бирор бир курсга ёзилиш.

3-Фаолият Диаграммаси (Activity Диаграм)

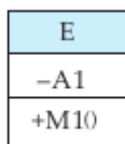
Фаолият диаграммасида тизимнинг турли хил компоненталари орасидаги вазифалар оқими тасвирланади.

4-Жорий этиш Диаграммаси (Имплементатион Диаграм)

Ушбу диаграмма тизимнинг барча ҳам дастурий ҳам қурилмавий амалга оширилиши даражасини тақдим этади.

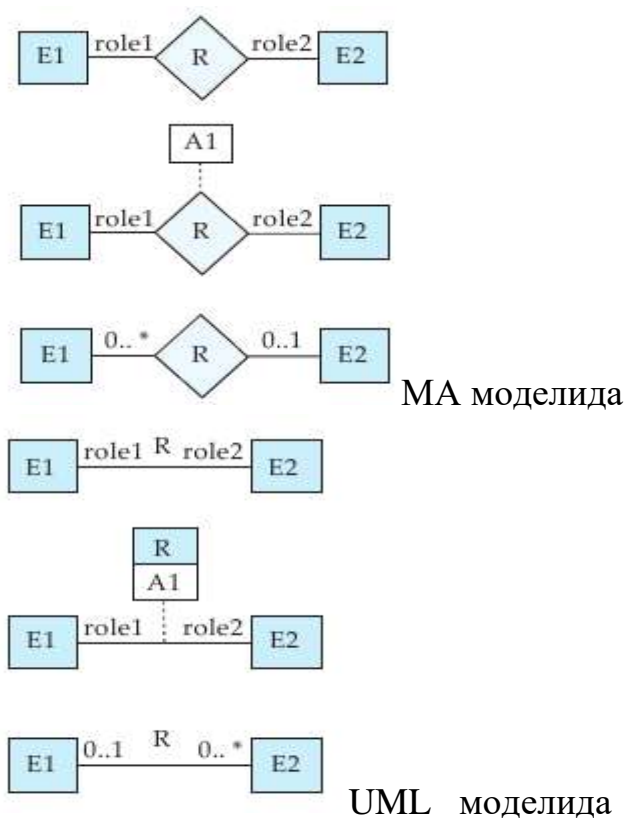


МА моделидаги атрибутларни берилиши



Тенг ҳолатда UML да берилиши

Моҳият-Алоқа модели ва UML да икки томонлама боғланиш, атрибутларни боғланиши, амаллардаги чекловлар .



3.5. Реляцион маълумотлар базаси дизайни²²

Умуман реляцион маълумотлар базаси дизайнининг мақсади маълумотларни кераксиз резервлашларсиз сақлаш ва маълумотни осон топа олишга ёрдам берувчи реляцион схемалар комплексини генерация қилишдир. Бу ўзига мос нормал формадаги схемаларни дизайнини яратиш орқали бажарилади. Қайси схема бизга керакли нормал формаларнинг бири эканлигини аниқлаш учун, биз маълумотлар базаси орқали моделлаштираётган реал-дунё предмет соҳаси ҳақида маълумот керак бўлади. Бу маълумотларнинг баъзилари яхши дизайнлаштирилган E-R диаграммада мавжуд бўлади, аммо предмет соҳа ҳақидаги қўшимча маълумот ҳам керак бўлиши мумкин. Бу бўлимда биз функционал боғлиқликлар тушунчасига асосланган реляцион маълумотлар базаси дизайни учун формал методни таништирамиз. Биз кейин функционал боғлиқликлар ва маълумот боғлиқликларининг бошқа турлари нуктаи назаридан нормал формаларни топамиз. Биринчидан биз берилган моҳият-алоқа дизайнидан олинган схемалар нуктаи назаридан реляцион дизайн муаммосини кўрамиз.

E-R модел объект ва муносабатлар тўпламига маълум даражали структура остига эга атрибутларга эгалик қилиш имконини беради. Биз кўп

²² 1. Silberschatz, Abraham. Database system concepts // pages-257-260

қиймат атрибутларга эга E-R модели жадвал яратганимизда, ундаги структура ости қисмини батараф етамиз. Ўўшма атрибутлар учун ҳар бир компонента ўз ҳуқуқини кўра атрибут бўлишига қўйиб берамиз. Кўп қийматли атрибутлар учун биз кўп қийматли тўпламнинг ҳар бири учун бир кортеждан яратамиз.

Реляцион моделда биз ушбу ғояга шакл берамиз. Домен бу агар домен элементлари бўлинмас бўлса. Реляцион схема биринчи нормал форма деймиз, агар ҳамма R нинг атрибутлар доменлари атомлар бўлса.

Номлар тўплами бу но-атомик қийматга мисол бўла олади. Ўша атрибутлар, масалан адресс атрибути ҳам но-атомик доменлардир, чунки бу атрибутда кўча, шаҳар, давлар ва зип кодлар каби элементларга бўлинади.

Бутун сон(integer) лар атомлардир, шунинг учун ҳам бутун сонлардан ташкил топган тўплам ҳам атомик доменларга киритса бўлади, лекин ҳамма бутун сонлар тўпламининг тўплами эндиликда атомик домен бўла олмайди. Фарқи шундаки, биз бутун сонлар қандайдир қисм остиларга эга эмас деб ҳисоблаймиз, лекин бутун сонлар тўплами қисмларга (тўплам элементларга) эга деб ҳисоблаймиз. Муҳими домен ўзи нима эканлиги эмас, балки биз қандай қилиб домен элементларини ўз базамизда ишлатишимиз муҳимдир.

Юқоридаги фикларга яхшироқ тушунишингиз учун қуйидагича мисол келтирсак. Тасаввур қилин ташкилот ишчиларнинг идентификацион номерини киритаётган бўлсин: биринчи иккита ҳарф департаментни билдирса қолган тўрт рақам ўша департаментдаги ишчининг уникал рақамидир. Масалан “CS0001” ва “EE1127”. Бундай идентификацион рақамлар кичикрок бўлимларга бўлинади, шунинг учун улар atomic эмаслар.

Бундай идентификацион рақамлар ишлатилса, унда ишчининг департаментини билиш учун ушбу идентификацион рақам структурасини ажратиб чиқадиган код ёзиш керак бўлади. Бунинг учун қўшимча дастурлашишлари лозим ва информация дастурий даражада амалга оширилади базада эмас. Агар бу рақамларга Primary key ишлатилинса муаммолар кўпаяди: агар ишчи департаментини ўзгартирса унинг идентификацион рақами базада ҳамма ерда ўзгариши лозим бу эса осон бўлмайди, дастур унга қайта идентификацион рақам генерация қолиши лозим, бўлмаса ушбу рақамни интерпретация қиладиган код хатолик чиқаради.

Тепадаги мулоҳазадан бизнинг курс идентификаторимиз “CS-101” бунда “CS” Computer Science департаменти доменимиз атомик эмасдек туюлади. Бу албатта инсонлар учун но-атомик бўлса ҳам база учун домен атомик ҳисобланади, чунки база application ни бўлишга ҳаракат қилиб

департамент бўлагини брeвиатурасини олиб бермайди, буни алоҳида инсонлар томонидан киритилган код бажаради.

Функционал боғлиқликларни ишлатиш орқали декомпозициялаш

Бундан олдин биз иккинчи бўлимга киришдаги гапларни хулосалаб:

Умуман олганда биз атрибутлар тўплами учун Грек ҳарфларини ишлатамиз. Биз реляцион схема учун кичик Роман ҳарфи кетидан қавс ичида ката Роман ҳарфини ишлатамиз ($r(P)$). Бунда схемар алоқа учун ишлатилинади, R эса атрибутлар тўпламини билдиради. Албатта реляцион схема бу атрибутлар тўпламидир, лекин ҳамма атрибутлар тўплами ҳам схема бўла олмайди. Кичик Грек ҳарфлари орқали биз атрибутлар тўпламини номлаймиз у схема бўлиши ҳам бўлмаслиги ҳам мумкин.

Атрибутлар тўплами суперкей бўлса биз уни K деб белгилаймиз. Super key хусусий реляцион схемага тегишли, шунинг учун биз “ K $r(P)$ нинг super key” деган термини ишлатамиз.

Биз алоқалар учун кичик ҳарфларни ишлатамиз. Уларни тушунарлироқ номлаб кетиш тавсия қилинади. Тариф ва формулаларда эса биз уни r деб номлаб кетганмиз.

Калитлар ва Функционал Боғлиқликлар. База ҳақиқий ҳаётдаги объектлар ва муносабатлар тўплами моделлаштиради. Одатда ҳақиқий ҳаётда маълумотлар бир қанча чекловларга (қоидаларга) эга.

Масалан:

1. Талаба ва инструкторлар уникал ID рақамларига эга.
2. Ҳар бир талаба ва instructor фақатгина битта исмга эга.
3. Ҳар бир талаба ва инструктор фақат биттагина департаментга қарайди.
4. Ҳар бир департаментга биттадан бино қарайди ва ўз бюджетига эга.

Super key таъриф: келин $r(P)$ реляцион схема бўлсин. R нинг K қисм тўплами $r(P)$ нинг Super key дейилади агар r даги t_1 ва t_2 кортежларда $t_1 \neq t_2$ бўлса, $t_1[K] \neq t_2[K]$ бўлади.

Super key атрибутлар тўплами бўлиб бутун кортежни ифодаласа, функционал боғлиқлик аниқ бир атрибут учун чеклов(қоида) бўлиб хизмат қилади. Масалан $R(P)$ реляцион схема берилган бунда $\alpha \subseteq R$ анд $\beta \subseteq R$.

R нинг экземпляр берилган, биз экземпляр $\alpha \rightarrow \beta$ функционал боғлиқликни қаноатлантиради деймиз агар, барча кортеж жуфтлар t_1 ва t_2 учун $t_1[\alpha] = t_2[\alpha]$ бўлса $t_1[\beta] = t_2[\beta]$ бўлиши лозим.

Функционал боғлиқлик супер бу ифода эта олмайдиган чеклов (қоида)

ифодалайди. Агар схема қуйидагича бўлса:

instdept (ID, name, salary, dept name, building, budget)

биз бу ерда функционал боғлиқликни икки хил усулда ишлата оламиз:

1. муносабатлар экзепларларини тест қилади булар берилган функционал боғлиқликни қаноатлантирадими йўқми шуни тест қилишимиз мумкин.

2. чеклов(қоида)ларни қонуний муносабатлар тўпламига жорий қилиш учун.

Қуйидагир R муносабатни кўрайлик. Бунда $A \rightarrow C$ ни қаноатлантиради.

Бунда A а1 қийматига эга, C эса с1, худди шундай икки кортеж A а2 ва C эса с2 қийматига эга. Lekin $C \rightarrow A$ Функционал боғлиқликни қаноатлантирмайди, чунки $t1=(a2, b3, c2, d3)$ ва $t2=(a3, b3, c2, d4)$. Ушбу кортеж бир хил C нинг с2 қийматига эга лекин ҳар хил A нинг а2 ва а3 қийматларига ҳам эгадир.

Бир хил функционал боғлиқликлар тривиал дейилади чунки улар ҳамма муносабатлар томонидан қаноатлантирилади. Масалан $A \rightarrow A$ Аатрибутга эга барча муносабатлар томинидан қондирилади.

Маълумотлар базаси асосан MB нинг лойиҳасидан таркиб топади. Моҳият-алоқа модели MB ни яратиш мақсадида энг кўп қўлланиладиган модел ҳисобланади. Ушбу модел ўзида маълумотларни, маълумотлар орасидаги боғлиқликни ва амалларни намоёиш этади.

Моҳият- бу реал ҳаётда мавжуд ва бошқаларидан хусусиятига кўра фарқланувчи объектдир. Биз моҳиятни унга таъриф берувчи атрибутлар ёрдамида аниқлаб киритамиз.

Боғлиқлик-бир қанча моҳиятлар орасидаги ўрнатилган алоқа. Боғлиқликларни ўрнатишда бир хил турдаги алоқаларнинг тўламиндан фойдаланамиз. Худди шунингдек моҳиятларни ўрнатишда ҳам бир хил турдаги моҳиятлар йиғиндисидан фойдаланилади.

Асосий чизмада бу бир қанча моҳиятлар сони бир-бирлари билан ўзаро қандай алоқада боғланганлиги тасвирланади.

Моҳиятни ўрнатилишида бошланғич калитнинг етарлича атрибутлари мавжуд эмаслиги кучсиз ўрнатилган моҳият деб номланади. Бошланғич калит мавжуд бўлган моҳият эса кучли моҳият дейилади.

Моҳият алоқа модели турли қисмлари MB си дизайнерларига модел уни янада ўзгача ҳолада тасвирлашнинг кўпгина усулларини тақдим этади. Бунда ҳам асосан ғоя ва объектлар, атрибут ва бўғланишлар моҳият

томонидан белгиланади. Берилган объект структурасининг барча кўриниши моҳиятнинг кучлилики даражаси орқали тушинтирилади.

МВси лойиҳаси моҳият-алоқа моделининг реляцион схемалари тўплами ёрдамида намоён этилади. Бунда МВ да ҳар бир моҳият ва ҳар бир боғлиқлик бутун бир бирлашган реляцион схема асосида ўрнатилади. Бу формалар моҳият-алоқа моделидан реляцион схемани ажратишга асос бўлиб хизмат қилади.

Махсулаштириш ва умумлаштириш юқори даражадаги моҳият билан қуйи даражадаги моҳиятни ўрнатишлар ўтасида алоқага аниқлик беради. Умумлаштириш бу бир ва ундан ортиқ қуйи даражадаги моҳиятлар ўзидан юқори даражадаги моҳиятларни умумий йиғилган натижасидир. Юқори даражадаги моҳиятнинг атрибутларини ўрнатиш қуйи даражали моҳиятларни ўрнатиш каби бир хилда бўлади.

Гуруҳлаш бу боғлиқликларда қатнашиши мумкин бўлган ва юқори босқич моҳиятларни ўрнатишдагидек боғлиқликларни ўрнатишдагидек мавҳум амалдир.

UML - бу машҳур моделлаш тилидир. УМЛ синф диаграммаси синфларни моделлашда кенг қўлланилиб маълумот моделларида умумий мақсад учун фойдаланилади.

Назорат саволлари

1. UML нима?
2. Лойиҳалаштириш жараёни деганда нимани тушунаси?
3. Қарор қабул қилишда лойиҳаланишнинг ўрни ҳақида гапириб беринг?
4. Маълумотларнинг муносабати деганда нима тушунаси?
5. Стандарт лойиҳалаш тили нима учун керак?
6. МА ва UML нинг фарқи нимада?
7. Реляцион маълумотлар нима?
8. Дизайнни яратиш усуллари?
9. ER моделни яратиш усули?
10. Калит атрибутлар нима учун керак?

Фойдаланилган адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. см. 2011. ISBN 978-0-07-352332-3, chapter 1-2, page 57-112.
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).

3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)" (2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000).
5. <http://www.colapublib.org/periodicals/>
6. <http://houstonlibrary.org/research/databases>

4 – мавзу: Сўровлар ва уларни қайта ишлаш босқичлари (2 соат).²³.

Режа:

- 4.1. Сўровларни қайта ишлашнинг босқичлари.
- 4.2. Сўровни баҳолаш ўлчовлари.
- 4.3. Танлаш амали.
- 4.4. Саралаш ва бирлаштириш амаллари.

Таянч иборалар: *Сўров, реляцион алгебра, танлаш, оптималлаштириш, баҳолаш, саралаш, излаш, бирламчи ва иккиламчи калит.*

4.1. Сўровларни қайта ишлашнинг босқичлари

Сўровларни қайта ишлаш – маълумотлар базасидан маълумотларни олиш учун фойдаланиладиган асосий вазифалар сирасига кирадиган амаллардан бўлиб ҳисобланади. Бу жараён юқори босқичли дастурлаш муҳитларида шакллантириладиган сўровларни МВ учун мослаштириш усуллари (чора тадбирларини), акслантириш, оптимизациялаш ва сўровларни баҳолаш тизимларини қамраб олади.

Сўровларни қайта ишлашнинг асосий қадамлари 4.1 – расмда келтирилган.

Асосий қаламдлар қуйидагилардан иборат:

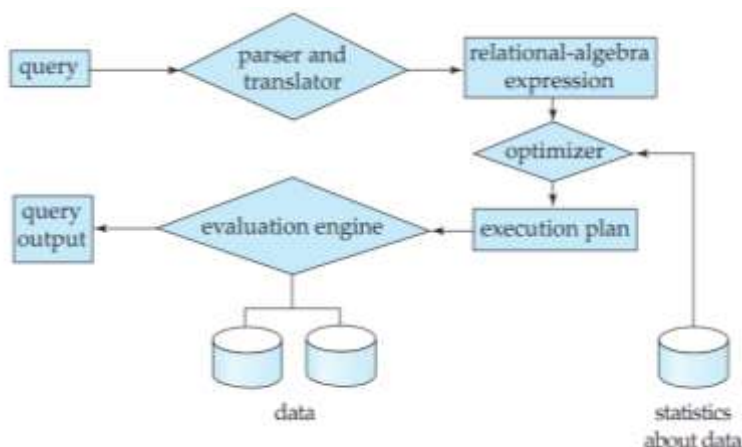
1. Таҳлил ва таржима;
2. Оптималлаштириш;
3. Баҳолаш.

Биринчи навбатда сўровни фойдаланиш мумкин бўлган кўринишга келтириш керак. Бунинг учун SQL тилидан унумли фойдаланиш мумкин. Аммо бу тил инсонлар учун баъзи бир муаммоларни келтириб чиқаради. Тизимнинг ички сўров тузилишига келтириш асосий муаммо бўлиб ҳисобланади. Бунинг учун фойдали бўлган асосий усуллардан бири бу кенгайтирилган реляцион алгебра асосларидир.

Шундай қилиб биринчи вазифада тизим сўровни қайта ишлаб, уни тизимнинг ички формал кўринишига келтириши керак. Бу жараён худди бошқа тизимлардаги парсер (parser) компилятор каби амалга оширилади. Сўровнинг ички формал кўринишини шакллантириш учун фойдаланувчининг сўровини таҳлиловчи модул ёрадамида синтаксис таҳлил қилиши керак. Сўровларнинг кўпчилиги МВларнинг муносабатлари билан

²³ 1. Silberschatz, Abraham. Database system concepts // pages-537

боғлиқдир. Модул сўровнинг тахлилидан сўнг унинг шажарасини кўринишда тасвирлашди ва уни реляцион алгебра амалларига ўтказди. Агар сўровнинг тузилиши тўлиқ реляцион алгебра амаалри билан бир хиллик маънони сақлаб қолиши зарур. Жуда кўплаб комплияторлар тахлиллашнинг турли хил усулларидан фойдаланилади.



4.1- расм. Сўровларни қайта ишлашнинг асосий қадамлари.²⁴

Жавобларни олиш учун сўровга боғлиқ ҳолда жуда кўп қоидалар, турли хил усуллар мавжуд. Масалан, SQL да тузиладиган сўровлар ҳар хил амаллар ва белгиларни олиши мумкин. Ҳар бир SQL сўров ўзга усул ёрдамида реляцион алгебра кўринишида ўтказишнинг бир нечта усуллари мавжуд. Бундан ташқари реляцион алгебра сўровларнинг амалий баҳоланиши имконияти билан ҳам боғланади.

Қуйидаги сўровни кўриб чиқамиз

```

select salary
from instructor
where salary < 75000;
  
```

Бу сўров реляцион алгебрада қуйидагича шакллантирилади.

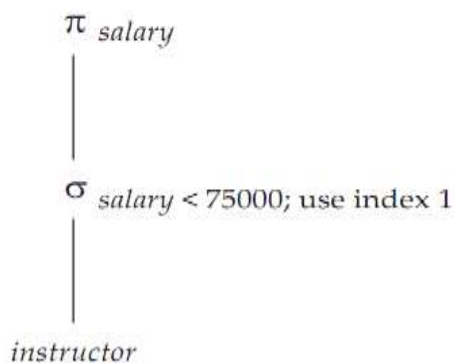
$$\bullet \sigma_{salary < 75000} (\Pi_{salary}(\text{instructor}))$$

$$\Pi_{salary}(\sigma_{salary < 75000}(\text{instructor}))$$

Бундан ташқари биз реляцион алгебра амалларини бир нечта ҳар хил алгоритмлар билан амлага оширишимиз мумкин. Масалан, мазкур танлашни амалга ошириш учун ҳар бир котреж(ёзув)ни ойлик маоши 75000 дан камли бўйича излашимиз мумкин. Агар ойлик атрибути V^+ шажараси индексида киритилган (рухсат берилган) бўлса, керакли ёзувни излаб топиш осон амалга ошириш мумкин.

²⁴ 1. Silberschatz, Abraham. Database system concepts // pages-538

Сўровни тўлиқ баҳолаш учун фақатгина реляцион алгебранинг муносабларидан фойдаланиш етарли эмас, бунинг учун ҳар бир берилган амални баҳолаш инструкция ёзиш орқали амлага ошириш мумкин. Инструкция ёзишда алгоритмларда махсус амалларга эътибор қаратиш ва улардан фойдаланиш учун аниқ бир ёки бир нечта индекслардан фойдаланиш тавсия этилади. Реляцион алгебранинг амаллари инструкция аннотация билан келтирилишини оддий баҳолаш мезони сифатида олиниши мумкин. Оддий амаларнинг ўзаро боғлиқлиги сўровларни баҳолашда ишлатилиши мумкин ва ўзи билан сўровнинг режаси ёки баҳолаш режасини ҳам амлга оширилиши назарда тутилган бўлади. 4.2 расмда Бизнинг юқорида келтирган сўровнинг баҳолаш режаси кўрсатилган. Аниқланган индексда (расмда Индекс 1) бир қатор амаллар танлаш мумкин. Сўровнинг бажарилиши сўровни баҳолаш режаси орқали бажарилиб сўровга жавоб қайтаради.



4.2 – расм. Сўровни баҳолаш режаси.

Берилган сўровларни баҳолаш учун турлича вақт сарфланиши мумкин. Биз фойдаланувчидан унинг сўровини аниқ ва маъноли қилиб ёзишини талаб қилишимиз орқали сўровни баҳолаш жараёнини тезлаштириш ва самарали амалга оширишимиз мумкин. Сўровни баҳолаш режасига ажратиладиган вақтни кичиклаштириш сўровнинг режасини тузишга ҳам боғлиқдир. Бу масалалар Сўровларни оптималлаштириш масаларига олиб келади. Бу масаларни кейинроқ батафсил кўриб чиқамиз.

Сўровнинг режаси танланиши билан сўров шу танланган режа асосида баҳоланиди ва олинган натижа экранга чиқарилади.

Сўровларни қайта ишлаш учун ёзилган қадамларнинг ўзаро боғлиқлиги асосий омил бўлиб ҳисобланади. Ҳамма МВ ҳам бундай аниқликга асосланган қадамларга таяна олмайди. Масалан, реляцион алгебранинг қоида ва ифодаларидан фойдаланиш учун баъзи бир МВларида SQL сўровнинг тузилишида статик шажаралар мажмуасидан фойдаланилади.

Аммо бу ёзиладиган қоида МВларида сўровларни қайта ишлашнинг асосини ташкил қилади.

Сўровларни оптималлаштириш учун сўровларни оптималлаштирувчи ҳар амалга ажратилган вақтни олдиндан билиши шарт. Аммо бу ажратиладиган вақтни баҳолаш муҳим ва машақатли вазифа ҳисобланади. Чунки, бу иш турли параметрлар, маълумотларни қайта ишлашга мўлжалланган хотира воситалари билан боғлиқ ва ҳар бир амални бажарилишини баҳолаш имкониятини бериш зарур. Бу мавзуда сўровни баҳолаш режасида ҳар бир амални баҳолашни, сарфланадиган вақтни ўрганамиз. Оптимизация учун кейинроқ тўхталамиз.

4.2. Сўровни баҳолаш ўлчовлари.²⁵

Сўровни баҳолаш учун бир нечта режалар мавжуд бўлса ҳам муҳими шундаки уларнинг баҳоланиш нуқтаи назаридан таққослаш имкониятини бериш ва энг самаралисини танлаш имкониятини бқрини керак. Бунинг учун баҳолаш амалларида ҳар бир амалнинг баҳоланиши ва уларни йиғиш орқали сўровнинг баҳолаш режасини баҳолаш имконияти бўлади. Шундай қилиб, биз алгоритмдаги ҳар амаланинг баҳолашни, қисқа мазмунини келтиришни ўрганиш билан шуғулланамиз. Сўровнинг баҳосини баҳолаш МВсида маълумотлар билан, дискларга мурожаат сонлари, сўровни амлага ошириш учун сарфланадиган вақт, МВсининг тақсимланганлиги ва параллелиги, маълумотларнинг мантиқий боғланганлиги билан боғлиқ бўлиб ҳисобланади.

Жуда кўп МВсида маълумотларнинг тақсимланиши, техник воситаларда маълумотларни қайта ишлаш усуллари турлича амалга оширилади. Бизда маълумотларни қайта ишлаш алгоритмларни модификациялаш орқали уларни ишлаш тезлигини оширишимиз мумкин. Аммо техник воситалар билан ишлаш тезлигини ошириш бизга боғлиқ эмас. Шунинг учун сўровни оптималлаштиришга кетадиган вақтда техник воситаларининг вақтлари олдиндан аниқ ва улар ўзгармасдир. Берилган масалани паст босқичли амалларига сарфланадиган вақт тўлиқ очиб берилишига қарамай, масалани ечиш вақтини реал оптималлаштириш масаласи мушкул иш. Шунинг учун ҳам реал ҳаётда сўровларни оптималлаштириш марказий процессор CPU диққатида бўлади.

Сўровни баҳолаш режаси баҳолаш дискдан маълумотларни узатиш блоклар сони ва дисклар сони билан ўзаро боғлиқ ҳисобланади. Агар дискдаги блокка маълумот узатиш подсистемасининг ўртача вақти T_t ва блокка рухсат вақти T_c бўлса, унда b блокка маълумот узатиш ва s блокка

²⁵ 1. Silberschatz, Abraham. Database system concepts // page-540

рухсат вақтининг умумий кўриниши $b * T_t + s * T_c$ ҳисобланади. Техник воситалар учун T_c ва T_t ларни бугунги энг сўнги технологиялари асосида $T_c = 4 ms$ ва $T_t = 0.1$ деб танланса, 4 кБли блок ва маълумот узатишниш тезлиги секундига 10 МВга тенг бўлади.

4.3. Танлаш амали²⁶

Сўровларни қайта ишлашда файлни сканерлаш (ўқиш) маълумотларга муружаатнинг муҳим босқичи бўлиб ҳисобланади. Файлларни сканерлаш махсус излаш алгоритмлари билан фиксирланади ва танланади. Реляцион муносабатли МВсида файл сканерлаш бир файлга жойлашган барча муносабатларни кўздан кечириб чиқиши лозим.

Бир файлда муносабатли кортежларни сақлаган ҳолатида танлаш амалини кўриб чиқамиз. Энг оддий танлаш амалининг бажарилиши куйидагича амалга оширилади:

A1 (чизиқли излаш). Чизиқли излашда, тизим сўров шартини қониқтирадиган вариантларни кўриш учун файлнинг барча блокларини сканерлайди ва барча кортежларни текширади. Биринчи ҳаракат файлнинг биринчи блокига рухсат олиш билан бошланади. Агар биринчи блокдаги маълумот арашалган бўлса, кейинроқ қайта ишлаш учун уни қайд қилиб олинади. Танлашда бундай усуллари бошқа усулларга нисбатан жуда тез ва осон амалга оширилиши мумкин. Чизиқли излаш алгоритми ихтиёрий тартибда келтиришган файлларга нисбатан ишлатилиши мумкин.

Таклиф қилинадиган алгиритмлар мураккаб ва чизиқли излаш алгоритмиги нисбаттан самарали ҳисобланади. Излаш алгоритмлари ёрдамида сканерланган файлларни кўп ҳолларда индекслар деб номлашади. Биз танлаш предикатидан фойдаланамиз. Сўровни қайта ишлаш учун индексларни танлашни киритиш керак. Индекслар тизимидан фойдаланиладиган алгоритмларга куйидагилар киради.

A2(бирламчи индекс, калитга тенглик). Одатда бу излаш усули калит атрибут ва бирламчи калит билан солиштириш орқали амалга оширилади. Буни бир ёзувни танлаш учун ишлатиш мумкин. 4.1 жадвалга қаранг.

A3 (бирламчи индекс, калитга тенг эмаслик). Биз бирламчи калит билан бир нечта ёзувларни танлашимиз мумкин қачонки маълумотларнинг бирламчи калит маълумотнинг калит атрибути бўлмаса. Олдин усул билан фақатгина бир нечата ёзувлар тўпламини танлашда фарқ қилади. Шунингдек маълумотларни излаш калит бўйича сараланган ва ўзаро кетма-кетликка асосланган. 4.1 жадвалга қаранг.

²⁶ 1. Silberschatz, Abraham. Database system concepts // page- 541

A4 (иккиламчи индекс, тенлик). Бу танлаш шартни тенглик бўлганда иккиламчи индекс асосида амалга оширилади. Бу жараён битта ёзув танлаши мумкин агар калитга нисбатан тенглик бўлса, бир нечта ёзувни танлаши мумкин агар индексланган майдон калит майдон бўлмаса.

Биринчи ҳолда A2га мос келади.

Иккинчи ҳолда ҳар бир ёзув бошқа бир блокларга боғланган бўлиши мумкин. Қайсики киритиш (чиқариш) амалларига олиб келади. Ҳар бир киритиш (чиқариш) амалларини бажариш излаш жараёни ва блокка бериш жараёнидан ўтади. Бу ҳолда кетадиган вақт $(h_i + n) \cdot (T_t + T_s)$ билан ҳисобланади. Бу ерда n блокка боғланган ёзувлар сони. Агар ёзувларнинг кўп қисми блокка боғланган бўлса, сўровни қайта ишлашга кўп вақт сарфланиши мумкин.

4.1-жадвал

	Алгоритм	баҳоси	изоҳ
A1	Чизиқ излаш	$T_s + b_r * T_r$	Биринчи қадамларидан b_r + блок аниқлаштириш. b_r файлдаги блоклар сони.
A1	Чизиқли излаш, калитга тенглаштириш	Ўртача қиймат $T_s + (B_r/2) * T_t$	Берилган шартни қаноатлантирганича изланади ва калитга моси топилгун кейин тўхтатилади.
A2	Бирламчи B^+ -шажарали индекс, калитга тенглаштириш	$(h_i + 1) * (t_r + t_s)$	h_i - индекснинг юқори қиймати. Излаш индекси киритиш чиқаришни танлаш учун шажаранинг қийматини 1 га оширади. Ҳар бир киритиш чиқариш амали блокка муҳожатни талаб этади.
A3	Бирламчи B^+ -шажарали индекс, калитга тенглаштирамаслик	$h_i * (T_t + T_s) + b * T_t$	Ҳар бир шажара босқичи учун излаш, биринчи блок учун излаш. Бу ерда b блоклар сони. Id калит орқали излаш мумкин бўлган ёзувларга асаоланган блоклар. Бу блоклар асоси ёзувлар

			билан боғланган ва уларни қўшимча излаш талаб этилмайди.
A4	Иккиламчи В ⁺ -шажарали индекс, калитга тенглаштириш	$(h_i + 1) * (t_r + t_s)$	Бирламчи калитга мос иккиламчи калит тенглаштириш
A4	Бирламчи В ⁺ -шажарали индекс, калитга тенглаштирамаслик	$(h_i + n) * (t_r + t_s)$	Бу ерда n мос келмаганлар сони. Бунда А3 ўхшаш индекслардан фойдаланилади. Ёзувлар бошқа блокка жойлашганлиги ҳам инобатга олинади. N нинг катта қийматларида бу излаш самарали бўлади.
A5	Бирламчи В ⁺ -шажарали индекс, таққослаш	$h_i * (T_t + T_s) - b * T_t$	А3 алгоритмнинг калитга тенг бўлмаган ҳоли.
A6	Иккиламчи В ⁺ -шажарали индекс, таққослаш	$(h_i + n) * (t_r + t_s)$	А4 алгоритмнинг калитга тенг бўлмаган ҳоли.

4.5. Саралаш ва бирлаштири амаллари²⁷

4.5.1. Саралаш амалари

Иккита масала юзасидан саралаш МВсида муҳим аҳамиятга эга. Биринчидан, берилган SQL сўров натижаси сараланган ҳолда чиқариш. Иккинчидан, сўровларни қайта ишлашда, баъзи бир муносабат каби реляцион амаларда самарали натижа бериши мумкин. Сортировканинг ҳам бир неча усуллари мавжуд

Муносабатларни сартировка қилишда калитларнинг индексларини яратиш усулларидан фойдаланамиз. Бундай муносабат фақат мантикий мазмунга эга. Саралаш масаласи кенг ўрганилган бўлиб ҳисобланади.

²⁷ Silberschatz, Abraham. Database system concepts // page-546

Улардан муносабатлар орқали саралаш ва техник воситалар билан боғлиқ саралашлар ҳам киради. Биринчи ҳол учун стандарт саралаш усулларидан фойдаланиш мумкин. Барча ҳолларда иккинчи ҳол учун қараймиз.

Ички Merge саралаш алгоритми.²⁸

Муносабатли саралашнинг бу алгоритм хотирада прогонка қилинмайдиган саралашга ички саралаш дейилади. Бундай алгоритмлар камдан кам ҳолларда ўзаро қўшилган алгоритмлар асосида фойдаланиб келинади. Ички қўшилган саралаш алгоритмини қуйидагича ёзиб олиш мумкин. Бунда М билан асосий хотирадаги блокларни сонини белгилаймиз. Агар уларга саралаш имконияти берилган бўлса.

1. Биринчи қадамда, сараланган трасс қатор яратилади, ҳар бир прогон сараланади ва керакли ёзув муносабатларини сақлайди.

$i = 0;$

repeat

***read** M blocks of the relation, or the rest of the relation,
whichever is smaller;*

***sort** the in-memory part of the relation;*

***write** the sorted data to run file R_i ;*

$i = i + 1;$

***until** the end of the relation*

2. Иккинчи қадамда прогон бирлаштирилади. Жорий вақтда N жами сон ва M битта блокни танлаш имкониятлари сони ва бир блокдаги прогонлар сони. Бу қадам қуйидагича амалга оширилади.

***read** one block of each of the N files R_i into a buffer block in memory;*

repeat

***choose** the first tuple (in sort order) among all buffer blocks;*

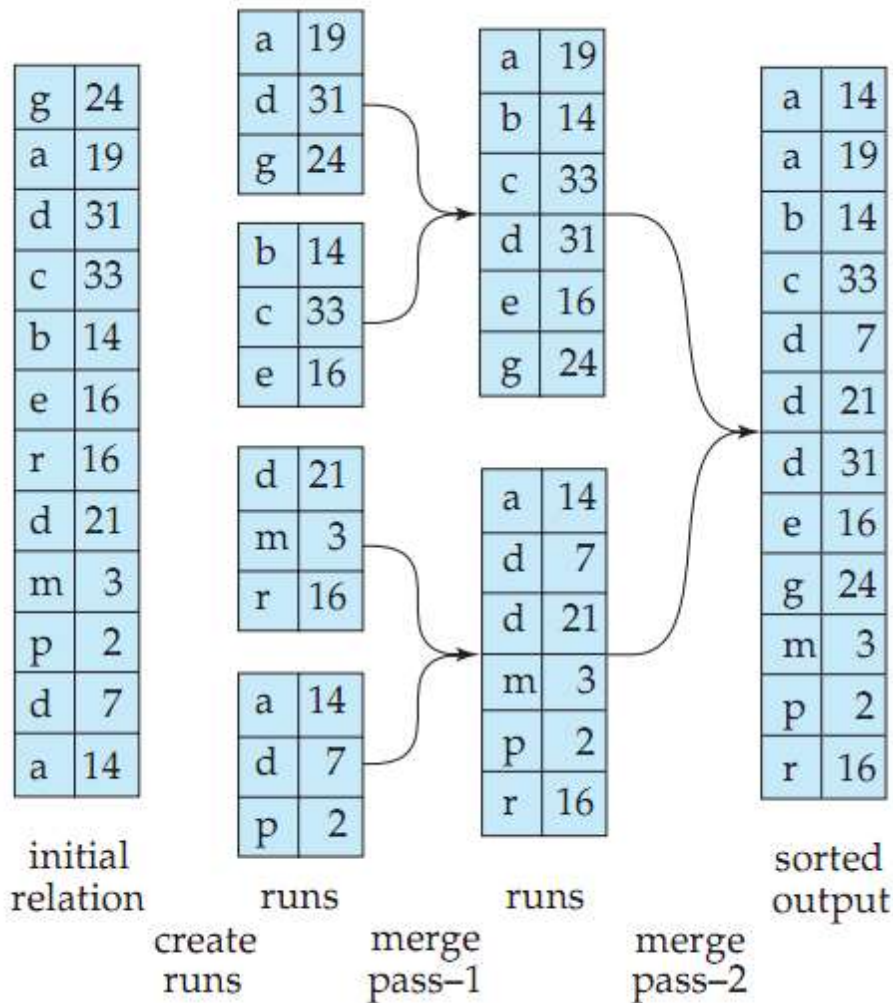
***write** the tuple to the output, and delete it from the buffer block;*

if the buffer block of any run R_i is empty and not end-of-file(R_i)

***then read** the next block of R_i into the buffer block;*

***until** all input buffer blocks are empty*

²⁸ Silberschatz, Abraham. Database system concepts // page-547



1.5. – расм. Ички Merge саралаш алгоритми.

Ички Merge саралаш алгоритмини баҳолаш.

Ички саралаш ва қўшилининг баҳолашни қуйидагича амалга оширамиз. b_r билан блоклар сони, r муносабатларни сақловчи ёзувлар сони.

$$b_r * (2 \left[\log_{M-1} \left(\frac{b_r}{M} \right) \right] + 1)$$

12.4 жадвалдаги баҳолаш алгоритмларни қўллаб, юқоридаги формулани қуйидагича ёзиб олиш мумкин.

$$2[b_r/M] + [b_r/b_b] (2[\log M - 1(b_r/M)] - 1)$$

Бирлаштириш амалари²⁹

Бу амаллар муносабатларни бирлаштириш ва баҳолашнинг алгоритмларига бағишланади. Биз қуйидагича тенг ўринли бирлаштиришни киритамиз.

²⁹ Silberschatz, Abraham. Database system concepts // pages-550-551

$$r \bowtie_{r.A=r.B} s$$

Бунда А ва В атрибутлар ёки атрибутлар тўплами бўлиб, муносабатлар билан боғланган бўлиши керак. Куйидагича мисол олиш мумкин.

student \bowtie *takes*

Олдинги бобларга келтирилган мисоллардан фойдаланиб, икки муносабатнинг қийматларини келтириш мумкин.

- талабаларнинг ёзувлар сони: $n_{student} = 5.000$
- талабанинг блоклар сони: $b_{student} = 100$.
- takes ларнинг ёзувлар сони: $N_{takes} = 10,000$.
- takes блокларининг сони: $b_{takes} = 400$.

Бирлаштирилган такрорланишлар.³⁰

Куйида келтирилган 1- алгоритмда тера боғланиш орқали ҳисоблашнинг оддий алгоритми келтирилган. Унда $s \bowtie_{\theta} r$ муносабат келтирилган бўлиб, r ва s лар ички муносабат ҳисобланади. Бу алгоритмлар ички алгоритмлар деб номланади.

```

for each tuple  $t_r$  in r do begin
    for each tuple  $t_s$  in s do begin
        test pair  $(t_r, t_s)$  to see if they satisfy the join condition  $\theta$ 
        if they do, add  $t_r \cdot t_s$  to the result;
    end
end

```

1- алгоритм бирлаштирилган алгоритм.

```

for each block  $B_r$  of r do begin
    for each block  $B_s$  of s do begin
        for each tuple  $t_r$  in  $B_r$  do begin
            for each tuple  $t_s$  in  $B_s$  do begin
                test pair  $(t_r, t_s)$  to see if they satisfy the join condition
                if they do, add  $t_r \cdot t_s$  to the result;
            end
        end
    end
end

```

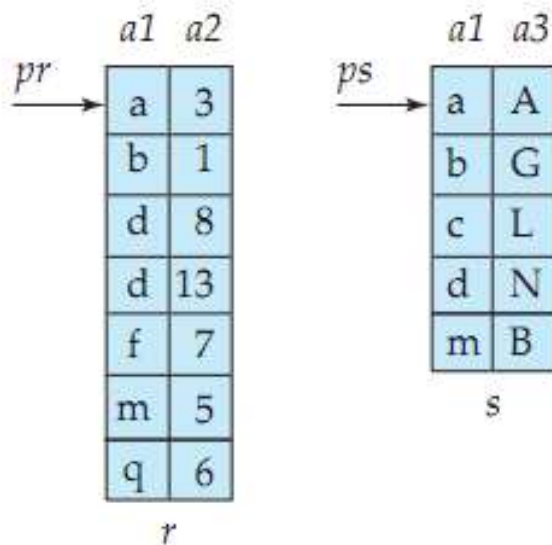
³⁰ Silberschatz, Abraham. Database system concepts // pages-550-551

2- алгоритм. Блокли бирлаштириш алгоритми

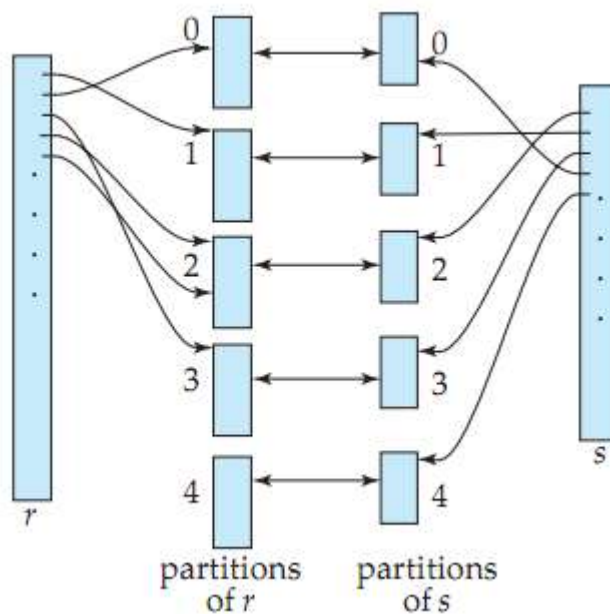
Merge-Join алгоритми

```
pr := address of first tuple of r;  
ps := address of first tuple of s;  
while (ps ≠ null and pr ≠ null) do  
  begin  
    ts := tuple to which ps points;  
    Ss := {ts};  
    set ps to point to next tuple of s;  
    done := false;  
    while (not done and ps ≠ null) do  
      begin  
        ts' := tuple to which ps points;  
        if (ts'[JoinAttrs] = ts[JoinAttrs])  
          then begin  
            Ss := Ss ∪ {ts'};  
            set ps to point to next tuple of s;  
          end  
          else done := true;  
        end  
      tr := tuple to which pr points;  
      while (pr ≠ null and tr[JoinAttrs] < ts[JoinAttrs]) do  
        begin  
          set pr to point to next tuple of r;  
          tr := tuple to which pr points;  
        end  
      while (pr ≠ null and tr[JoinAttrs] = ts[JoinAttrs]) do  
        begin  
          for each ts in Ss do  
            begin  
              add ts ⋈ tr to result;  
            end  
          set pr to point to next tuple of r;  
          tr := tuple to which pr points;  
        end  
      end  
    end  
  end.
```

Бу алгоритмда JoinAttrs $R \cap S$ ва $T_R \times T_S$, яъни ўзаро муносабатли бўлиб, бу ерда T_R ва T_S атрибутлар бўлиб ҳисобланади.



4.5 – расм. Merge-Join муносабати учун саралаш.
Hash-join алгоритми³¹



4.6-расм. Hash-join алгоритми

Бу келтирилган алгоритмлардан ташқари яна бир нечта хил алгоритмлар бор. Уларни 1 адабиётларнинг 554-599 бетларда батавсил маълумот ва кенгайтирилган вариантлари келтирилган.

³¹ 1. Silberschatz, Abraham. Database system concepts // pages-558

Назорат саволлари

1. Сўровларни қайта ишлаш усуллари имконияти санаб беринг?
2. Саралаш алгоритмлари фарқлари нимада?
3. Сўровни баҳолаш ўлчовлари ва усуллари?
4. Танлаш амали ва усуллари ҳақида гапириб беринг?
5. Сўровларни қайта ишлаш ниама учун керак?

Фойдаланилган адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. cm. 2011. ISBN 978-0-07-352332-3.
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)"(2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000).
5. <http://www.colapublib.org/periodicals/>
6. <http://houstonlibrary.org/research/databases>

5 – мавзу: Транзакция. Транзакция концепцияси ва модели (2 соат).

Режа:

- 5.1. Транзакция тушунчаси.
- 5.2. Транзакция концепцияси.
- 5.3. Транзакциянинг оддий модели.
- 5.4. Қўшилмаслик ва мустақамлик транзакция.
- 5.5. Транзакцияларни бошқариш операторлари.

Таянч иборалар: *транзакция, SQL, амаллар кетма – келиги, транзакция хусусиятлари, сақлаш структураси.*

5.1. Транзакция тушунчаси

Амалиётда тез-тез ишончли қоғози, пул ўтказиш ва олиш учун бир нечта амалларни берилгани бўйича маълумотлар тўплами аосида бажаришга тўғри келади. Маълумотлар тўпланидан фойдаланувчи бир рақами амални бажаришни ўша вақтда амалга ошириши керак. Масалан, маблағларини бир жойга кўчириш тўғрисидаги маълумотларни тўлиқ амалга ошириш учун амаллар тўплами ҳисобланади ва ягона операция қилинади. Ўшандан харидорлар чегараларда маълумотлар тўплами тартиби тўғрисида бир неча операциялар аниқ ҳисобланади. Булар операцияларни минглаб ёзиш, ёки ўша ишида келишиш тўғрисида бундай бажармаслик, ҳеч бир операцияни амалга оширмасликка тўғри келади.

Тўпламлардаги амаллар тўғрисида операциялар ягона мантиқий бир иш рақами билан номланади. Бундай маълумотлар тўпланининг тартиби қарзни таъминлаш керак ва тегишли бажарилиши тўғрисида келишувларга қарамай рад жавоблари бериши керак, ёки ҳеч бир тўғрисида. Бундай қарзни билиш учун ажариш тўғрисида битимларни ишида бундай усул тўғрисида ўзаро мувофиқлаштирилмаган. Бундай амалларни ва вазифаларни транзакция ҳисоблайди ва бошқаради.

5.2. Транзакция концепцияси³²

Транзакция ягона тизим(алгоритм)ни бажариш бўлиб, маълумотларни ўчириш, янгилаш, танлаш ва бошқа бир нечта амалларни бажаришни тақозо қилади. Транзакция фойдаланувчи дастурий таъминоти билан интеграция

³² 1. Silberschatz, Abraham. Database system concepts // pages-627

килинади. Унда фойдаланувчи юқори босқичли дастурлаш тиллари (масалан, C ++ ёки Java) ва SQL лардан фойдаланиши мумкин. МВси билан боғланиш учун JDBC ёки ODBC билан интергация амалга оширилади. Транзакция бир функция ёки event шаклида келиши мумкин. Транзакциядаги барча амаллар begin transaction ва end transaction оперторлари орасида бўлади.

Бу амаллар кетма кетли фойдаланувчига ягона амал каби тасвирланиши керак. Транзакция яхлит ягона ва бир вақтда фаол бажариладиган амалдир. Шунинг учун ундаги барча амаллар бажарилади ёки бажарилмайди. Шундай қилиб транзакция амаларни бажаришни бошлайди аммо МВга ўзгартиришлар кирита олмайди ва транзакция тугаши билан у томондан бажарилган барча амаллар амалга оширилади. Бу жараёни кўз олдимизга келтириш учун амалларнинг бажарилиши вақтида бирор бир амалда нолга бўлиниш ҳодисаси юз бериши мумкин ёки ОСда техник носозлик вужудга келиши мумкин. Булардан кўриниб турибдики, техник воситалар ва амалларга аниқлик киритиш қийин иш. Агар бу ҳодисалар рўй бериш эҳтимоли бўлса, улар МВнинг вақтинчалик сақлаш жойига сақланади. Бу жараёнда вақтинчалик хотиранинг свойствасини “**ALL or NONE**” қилиб қўйиш керак ёки автоматик амалга оширилади.

Бундан ташқари, транзакция яхлит битта амал бўлгани учун МВсидаги амаллар бир биридан алоҳида бажарилмади ёки алоҳида транзакция ҳисобланмайди. Транзакция вақтида буни фойдаланувчи сифатида тўғри тушиниш керак. Умуман олганда SQL амаллари ҳам МВсига алоҳида – алоҳида мурожалари мавжуд бўлгани учун транзакция бир нечта SQL операторларидан ташкил топади. МВларининг тизими бу операторларнинг ишлашини тўлиқ таъминлаб бериши ва бир вақтда МВсининг операторларини ҳам бажариши зарур. Бу хусусият ўзга ишларга **қўшилмаслик** (изоляция) деб номланади. Агар система транзакцияни тўлиқ бажарса ҳам, бу қўшилмаслик кичик мақсадда фойдаланилади. Чунки система транзакция бажарилаётганлиги хис қилиши керак. Шундай қилиб, транзакция ҳодисаси барча тасодифий авария ҳолатларда ҳам сақлаб қолиш имкониятини яратиш учун хизмат қилади. Шунинг учун бу хусусият **мустваҳкамлик** деб номланади.

Юқорида келтирилган учта хусусият орқали ташкил этилган транзакция МВ системаси билан идеал структуралашган ҳисобланади. Бу эса транзакция учун баъзи бир талабларни киритишни тақозо этади. Транзакция маълумотларни яхлитлигини сақлаши лозим, агар транзакция қўшилмаслик транзакцияси хусусияти билан бажарилса кетма-кетлик билан МВ системасида ўзининг олдинги ҳолатига келиши керак ёки транзакция

охирида кетма-кетлик асосида асл ҳолатига қайтиши керак. Маълумотларнинг яхлитлигини сақланиши учун транзакция SQL операторларининг бажарилган амалларини тўлиқ назоратини амалга ошириш керак. Бу қандай амалга оширилади? Бу дастур тузувчининг ёзган кодлари орқали амалга оширилади. Бу хусусият **изчиллик** (кетма кетлик) деб номланади.

Буларнинг барчасини амалга ошириш учун транзакциялар қуйидаги хусусиятларни бажаришлари керак:

- Валентлик (бошқа атрибутлар билан бирикиш ёки алмашилиш қобилияти). Транзакциянинг барча амаллар натижаси МВсида амалга оширилиши керак ёки бирор биттаси амалга оширилмайди.

- Изчиллик (кетма кетлик). Транзакция бажарилганда қўшилмаслик (яъни транзакция бажарилаётган вақтда паралел бирорта транзакцияни бажарилмасли) асосида МВсининг яхлитлигини тўлиқ сақлаш.

- Қўшилмаслик. Агар бир вақтда бир нечта транзакция бажарилиши керак бўлиб қолса, тизим T_i ва T_j транзакцияларнинг бажарилишини таъминлайди(гарантиялайди), бунда T_i транзакция бажарилган сўнг T_j транзакция бажарилади, ёки аксинча. Агар T_i ва T_j лар ўзаро боғлиқ бўлса, бу T_i ва T_j транзакциялар битта транзакция ҳисобланади. Шундай қилиб транзакция бошқа транзакция амаллари ҳақида маълумотга эга бўлмайди. Улар шу ҳолда паралел ишлаш имкониятига эга бўлади.

- **мустваҳкамлик.** Қачонки транзакция мувофақиятли амалга оширилса, у бажарган амаллар ва ўзгаришлар МВ системасига киритилади ва сақланади (бирор бир техник хатолик рўй берса ҳам).

5.3. Транзакциянинг оддий модели³³

SQL кучли ва мураккаб тил бўлиб ҳисобланади. Шунинг учун биз оддий МВсининг тилларидан фойдаланамиз. Бунинг учун қўшиш, ўчириш каби амалларини алмаштириб оламиз. Шунингдек маълумотлар устида бажариладиган амалларни оддий тилдаги арифметик амаллар билан алмаштириб оламиз.

Одатда ҳар бир транзакциянинг номлари лотин ҳарфлари билан номланиши талаб этилади. Маълумотларга мурожат қуйидаги амал билан амалга оширамиз:

- read(X), X ўзгарувчига МВсидан маълумотларни узатади, тезкор хотирадаги X маълумотга мурожат орқали ўқиш амаллари учун.

³³ 1. Silberschatz, Abraham. Database system concepts // pages-629

- $write(X)$, X ўзгарувчидан MV сига маълумотларни узатади, тезкор хотирага X маълумотни ёзиш амаллари учун.

Мисол учун қуйидагича транзакция берилган бўлсин:

```
 $T_i : read(A);$   
 $A := A - 50;$   
 $write(A);$   
 $read(B);$   
 $B := B + 50;$   
 $write(B).$ 
```

Бу транзакциянинг иш MV сидаги A маълумотга қийматидан 50 ни олиб танлаш ва B маълумот қийматига 50 ни қўшиб қўйишдир.

Сақлаш структураси³⁴

Транзакция орқали амалга оширилган амалларни MV си системасига яхлит сақлаш учун олдиндан транзакция ёрдамида қайта йўланалидиган маълумот структурасини билиш керак бўлади.

Биз биламизки MV сида маълумотларни сақлашнинг бир нечта методикалари бор. Энди булар қаторига яна битта сақлаш синфи критилади ва у стабил (узлуксиз, доимий) сақлаш деб юритилади. Улар қуйидагилар:

- **Узоқ мудатли сақлаш.** Энергия билан боғлиқ маълумотларни сақлаш оддатда тезкор хотирада амалга оширилади. Бунга мисол сифатида MV сининг кэш хотирасини ҳам олиш мумкин. Тезкор хотирадаги маълумотларга мурожатлар тез ва аниқ содир бўлади. Шунинг учун оддатда маълумотларни узоқ мудатли сақлаш воситаларидан фойдаланиш керак.

- **Энергияга боғлиқ сақлашлар.** Бундай тизимларга сақланадиган маълумотлар бирга сақланган ҳолда учрайди. Буларга мисол қилиб, иккинчи хотира ташқи хотираларни олиш мумкин. Шунингдек ташиб олиш учун мўлжалланган қўрилмаларда ҳам сақлаш мумкин. Юқори технологиялар асосида ушбу амал тез амалга оширилади. Мазкур ташқи сақлаш воситалари маълумотларнинг йўқотилишига сабаб бўлиши мумкин.

• **Узлуксиз (доимий) сақлаш.** Бу шундай сақлаш воситаси бўлиши керакки, ҳеч қачон маълумотни йўқотиш мумкинмас. Назарий жиҳатдан доимий сақлашни ташқил қилиш мумкин, аммо амалий жиҳатдан бунинг иложи йўк. Бунинг баъзи бир методлардан фойдаланиш мумкин. Бу усуллар маълумотларни йўқотишни камайтиришга хизмат қилади. Буни ташқил қилиш учун бир нечта маълумотларни бир нечта маълумот кўчириб юриш

³⁴ Silberschatz, Abraham. Database system concepts // pages-632

воситасидан фойдаланиш мумкин. Янгилашни шундай амалга ошириш керакки унда маълумотларни йўқолишига йул қўймаслик ва кафиллик бериш лозим.

5.4.Қўшилмаслик ва мустаҳкамлик транзакция³⁵

Биз юқорида таъкидлаганимиздек транзакция яхлит амалга оширилади ва яқунланилади. Бундай транзакция тугалланган транзакция саналади. Агар биз бунинг тўлиқ хусусиятини таъминламоқчи бўлсак, транзакция МВсининг ҳолатига таъсир кўрсатмаслиги лозим. Шундай қилиб ихтиёрий ўзгаришлар, транзакция тугатилганидан сўнг МВсига инобатга олиниши керак. Агар транзакция мувофақиятли амалга ошмаса, амалга оширилган баржа ўзгаришлар МВсидан олиб ташланиши керак. Бундай барча ўзгаришларни log файлларда ёзиб борилади. Барча ёзувлар идентификация қилинади. Бу ҳақида тўлиқ маълумот 16 бобда келтирилган.

Биз транзакциянинг мувофақиятли бўлишига ишончимиз комил бўлиши керак. Шунинг учун транзакциянинг абстракт моделни келтирамыз. Транзакция қуйидаги асосий босқичлардан иборат

- Фаоллик, бошланғич ҳолат; Транзакция бу ҳолатда тайёр статик хатоларсиз бўлиши лозим.

- Хусусий хол учун тугатилган. Хатолардан холи хол учун тўлиқ амалларни бажарилишини таъминлаш.

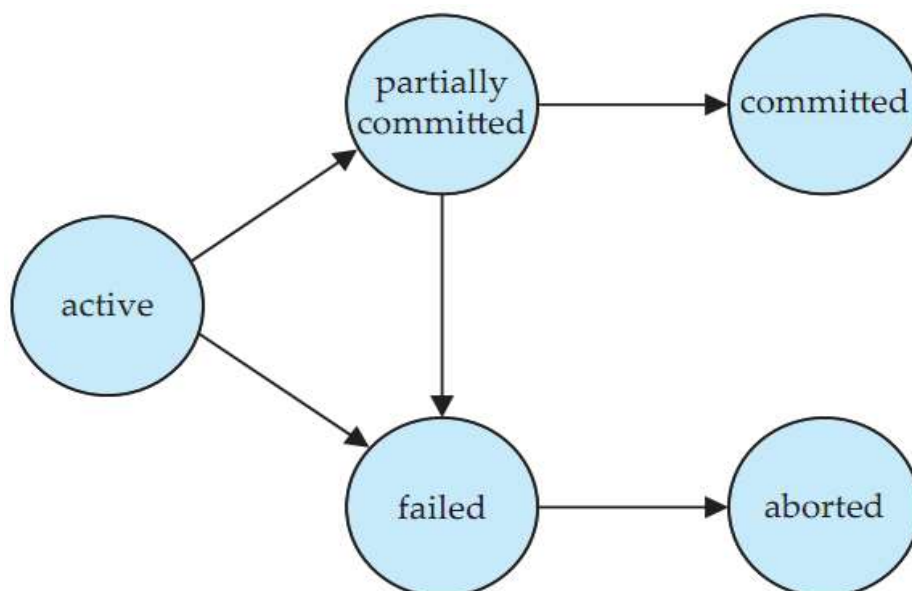
- Амлага ошмаслик (хатолик), Агар нормал иш фаолияти давом этириш мумкин бўлмаган ҳолат.

- Узулиш, агра транзакция бажарилаётган вақтда сбой бўлса, МВсини олдинги ҳолатга келтириш.

- Тугатилганлик, транзакция тўлиқ тугаганидан сўнги ҳолат.

Ушбу келтирилган босқичлар асосида абстракт моделни қуйидагича чизиш мумкин.

³⁵ 1. Silberschatz, Abraham. Database system concepts // pages-663



5.1-расм. Транзакциянинг асосий босқичлар модели.

Кўшилган транзакциялар

Фараз қиламиз бизда қуйидагича иккита T_1 ва T_2 транзакциялар берилган бўлсин. Бу транзакцияларни қўшилган вариантларини қуйидагича амалга ошириш мумкин.

T_1 транзакция

```

T1: read(A);
A := A - 50;
write(A);
read(B);
B := B + 50;
write(B).
  
```

T_2 транзакция

```

T2: read(A);
temp := A * 0.1;
A := A - temp;
write(A);
read(B);
B := B + temp;
write(B).
  
```

Бу T_1 ва T_2 транзакцияларини бир неча варианта биргаликда ишлатиш усуллари бор.

1. Кетма-кет ишлатиш принципи. Бунда аввал биринчи (иккинчи) транзакция сўнг иккинчи (биринчи) транзакция бажарилади.

T ₁	T ₂
<pre> T₁: read(A); A := A - 50; write(A); read(B); B := B + 50; write(B). </pre>	<pre> T₂: read(A); temp := A *0.1; A := A - temp; write(A); read(B); B := B + temp; write(B). </pre>

Ёки

T ₁	T ₂
<pre> T₁: read(A); A := A - 50; write(A); read(B); B := B + 50; write(B). </pre>	<pre> T₂: read(A); temp := A *0.1; A := A - temp; write(A); read(B); B := B + temp; write(B). </pre>

2. Маълумотларга мурожат асосида аввал T₁ транзакциянинг A маълумоти ва T₂ транзакциянинг A маълумоти, сўнг эса T₁ транзакциянинг B маълумоти ва T₂ транзакциянинг B маълумоти учун транзакцияни бажариш мумкин ёки аксинча аввал T₂ сўнг T₁.

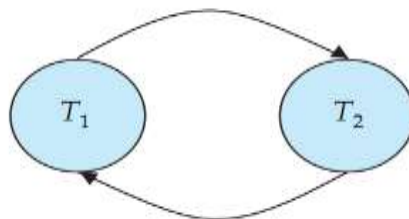
T ₁	T ₂
<pre> T₁: read(A); A := A - 50; write(A); read(B); B := B + 50; write(B). </pre>	<pre> T₂: read(A); temp := A * 0.1; A := A - temp; write(A); read(B); B := B + temp; write(B). </pre>

Бошқа вариантларда фойдаланиш хато маълумотларни келтириб чиқариши мумкии.

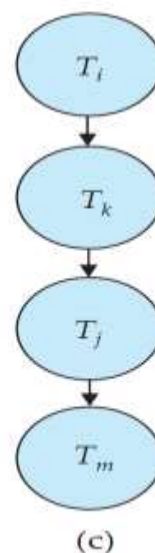
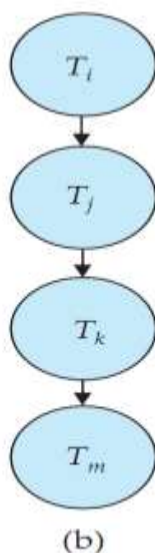
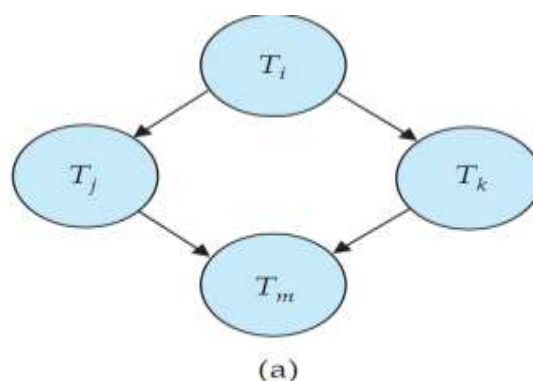
Қўшилган транзакцияларнинг графлар кўринишида тасвирлаш усуллари



5.2- расм. Кетма-кет транзакциянинг бажарилиши.



5.3- расм. Навбатма навбат транзакцияни бажариш.



5.5-расм. Топологик саралаш асосида транзакцияни бажариш.

5.5.Транзакцияларни бошқариш операторлари

	Оператор	Маъноси	Амал
1	COMMIT	Транзакцияни тугатиш	Транзакцияни ташкил қилувчи маълумотларни қайта ишловчи мураккаб ва ўзаро боғланган амаларни тугатиш
2	ROLLBACK	Транзакцияни бекор қилиш	Транзакция бажарилиши натижасида юз берган ўзгаришларни бекор қилиш
3	SAVEPOINT	Транзакция бажарилишида оралик нукта сақлаш	МВ ни оралик ҳолатини сақлаш. Бу кейинчалик шу ҳолатга қайтиш учун зарур бўлади.

Маълумотларни бошқариш операторлари • DCL (Data Control Language) – маълумотларни бошқариш

	Оператор	Маъноси		Амал
1	ALTER DATABASE	МВ ни ўзгартириш		Бутун маълумотлар базасига тегишли бўлган объектлар ва чекланишлар тўпламини ўзгартириш
2	ALTER DBAREA	МВ сақлаш соҳасин ўзгартириш		Аввал яратилган сақлаш соҳасини ўзгартириш
3	ALTER PASSWORD	Паролни ўзгартириш		Бутун маълумотлар базаси паролини ўзгартириш
4	CREATE DATABASE	МВ яратиш		Барча параметрларини кўрсатган ҳолда янги маълумотлар базасини яратиш
5	CREATE DBAREA	Сақдаш соҳасин яратиш		Янги сақлаш соҳасинияратиш ва унда маълумотлар жойлаштиришга рухсат этиши
6	DROP DATABASE	МВ ни ўчириш		Мавжуд маълумотлар базасини ўчириш (бундай амал бажариш ваколлатига эга бўлган фойдаланувчилар учун)
7	DROP DBAREA	МВ сақлаш соҳасин ўчириш		Мавжуд сақлаш соҳасини ўчириш (агар унда жорий вақтда фаол маълумотлар жойлашмаган бўлса)

8	GRANT	Ваколат бериш	МВ баъзи объектлари устида бир қатор амалларни бажариш ҳуқуқини бериш
9	REVOKE	Ҳуқуқдан маҳрум қилиш	Бирор объектга ёки объект устида бажариладиган баъзи амалларни бажариш ҳуқуқидан маҳрум қилиш

Назорат саволлари:

1. Транзакция нима?
2. Транзакцияни бошқаришнинг асосий босқичлари?
3. Транзакцияда сақлаш структуралри ҳақида гапириб беринг?
4. Қўшилмаслик ва мустақкамлик транзакциялари деганда нимани тушунасиш?
5. Транзакцияда фойдаланиладиган SQL операторларини санаб беринг?

Фойдаланилган адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. см. 2011. ISBN 978-0-07-352332-3.
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)"(2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000).
5. <http://www.colapublib.org/periodicals/>
6. <http://houstonlibrary.org/research/databases>

6 – мавзу: Маълумотларни йиғиш, жамлаш ва излаш усуллари (2 соат).

Режа:

- 6.1. Маълумотларни жамлаш тушунчаси.
- 6.2. Маълумотларни йиғиш тизими.
- 6.3. Маълумотларни сақлашни ташкиллаштириш.
- 6.4. Қоидалар омбори.
- 6.5. Ечимлар шажарасини қуришни классификациялаш.
- 6.6. Маълумотларни излаш усуллари.

Таянч иборалар: *қарор қабул қилиш, SQL пакетлари, маълумотларни сақлаш, қоидалар омбори, ечимлар шажараси, энг яхши бўлиниш, маълумотларни излаш, индекслаш, параметрли ва зонали индекслаш, долзарбликнинг вазли зоналари, машина ўқуви, термин частотаси ва вазни, тесқари маълумотли частота.*

6.1.Маълумотларни жамлаш тушунчаси

Корхоналар ишини ривожлантириш учун объектлар, уларнинг фаолияти ҳақида қарорлар, биржа ва миқозларни кўпайтириш тўғрисидаги маълумотлардан онлайн фойдаланиш жорий қилиш бугунги куннинг талабидир.

Бундай маълумотларни қайта ишлашнинг икки жиҳатлари бор.

Биринчи томони маълумот омбори асосида, бунда бир марказий омбор ичига бир неча манбаларда маълумотлар тўпланади. Маълумотларни сақлаш масалалари асосида маълумотлар билан муомалага киришишнинг бир қанча методларни ўз ичига олади. Яъни ноаниқ маълумотлар, баъзи хато маълумотлар, техника билан боғлиқ ва самарали натижа олиш билан боғлиқ бўлган маълумотларни қайта ишлаш ва фаоллаштириш усуллари киради.

Иккинчи омил бизнес масалалари учун асос бўладиган билим ва топилган маълумотлар йиғиндисини таҳлил қилишдир. Баъзи маълумотларни анализ қилиш турлари онлайн аналитик жараён (ОЛАП) учун SQL конструктураси орқали бажарилади.

Яна бир масаланинг ечими шуки, ахборотларни интеллектуал қайта ишлаш орқали билимларни ошириш, қайсики бир қанча турдаги маълумотларнинг катта ҳажмли намуналаридан фойдаланаган ҳолда. Маълумотларни қайта ишлаш турли хил мақсадли статистик маълумотларни

атамаларни қўшиб қўяди.

6.2.Маълумотларни йиғиш тизими³⁶

Маълумотлар базаси тадбиқи икки турдаги келишув -жараён ва қарорни қўллаш тизимларига классификация қилинади. Келишув-жараён тизимлари келишув ҳақидаги маълумотларни йиғиб боради, худди компаниялар учун сотиладиган маълумотларни ёки рўйхатга олиш усуллари ва университет учун қўшимча маълумотларни тўплаб боради. Келишув-жараён тизимлари бугунги кунда кенг миқёсда қўлланилмоқда ва бу тизимлар орқали бошқарилдиган маълумотларнинг кўп ҳажми ташкилотлар томонидан ҳисоб китоб қилинган. Қарорни қўллаш тизимларининг мақсади юқори даражадиги маълумотлар, келишув жараён тизимидаги йиғилган маълумотлар ва турли хил қарорларни қабул қилишдаги юқори даражадаги маълумотлардан фойдаланишни қўллаб қувватлайди. Қарорни қўллаш тизими дўконларни заводларда ишлаб чиқарилган маҳсулотлар билан таъминлаш ёки университет учун қабул қилинган аризаларни ҳақидаги қарорлар қабул қилишга ёрдам беради.

Масалан, компания маълумотлар базаси мижозлар ва келишувлар ҳақидаги энг муҳим катта ҳажмли маълумотлардан ташкил топган. Маълумотлар сақлаш ҳажми юзлаб гигабайтлар ёки ҳаттоки терабайтлар катта ҳажмда бўлишини талаб қилади.

Қарор қабул қилиш учун маълумотларни сақлаш ва излаш масалаларида бир нечта муаммоли масалар бор:

- Қарор қабул қилиш учун сўровларнинг кўпчилик қисми SQL тилида ёзилиши ёки SQL тилига ўтказишнинг имконияти йўқлиги. Шу ишларга мослаб бир нечта SQLнинг кенгайтирилган вариантлари яратилган.
- Сўров тиллари МВсидаги статистик маълумотларни детал таҳлил қила олмайди. Бунинг учун бир нечта пакетлар яратилган, масалан, SAS ва C++. Бу пакетлар МВсида SQLнинг имкониятларини кенгайтириб, статистик маълумотларни қайта ишлаш имконини беради. Жуда кўплаб соҳаларнинг статистик маълумотлари библиографик белгилар орқали амалга оширилади.
- Катта компаниялар бизнес ва маркетинглар учун маълумотларнинг катта ҳажмлари билан ишлайди. Шунинг учун маълумотлар турли структурада саланади. Бу ҳолларда SQL нинг кенгайтирилган пакетлари ҳам ожизлик қилади. Бунда сўровларни самарали бажариш

³⁶ 1. Silberschatz, Abraham. Database system concepts // 887 - бет

учун компаниялар махсус маълумотларни сақлаш базаларини яратадилар. Маълумотларни сақловчи база бир нечта усуллар ва алгоритмлар ёрдамида маълумотларни бир нечта МВсидан йиғиб олади. Шундай қилиб, бу фойдаланувчиларга ягона бир хил интерфейс асосида Маълумотларни сақлаш масаларини ўрганишга ёрдам беради.

- Билимларни аниқлаш усуллари, берилган тайёр форм дата билан билимларни аниқлашга ўтиш мумкин. Бунинг Data mining да билимларни аниқлар усуллари яратилган. Булар маълумотларни интеллектуал анализ қилиш имкониятини беради.

Қарор қабул қилиш. Бу катта жараён бўлиб, унда одатда терминлар асосида паст даражада қараш натижасида маълумотларни статик ва интеллектуал таҳлил қилиш назарда тутилади.

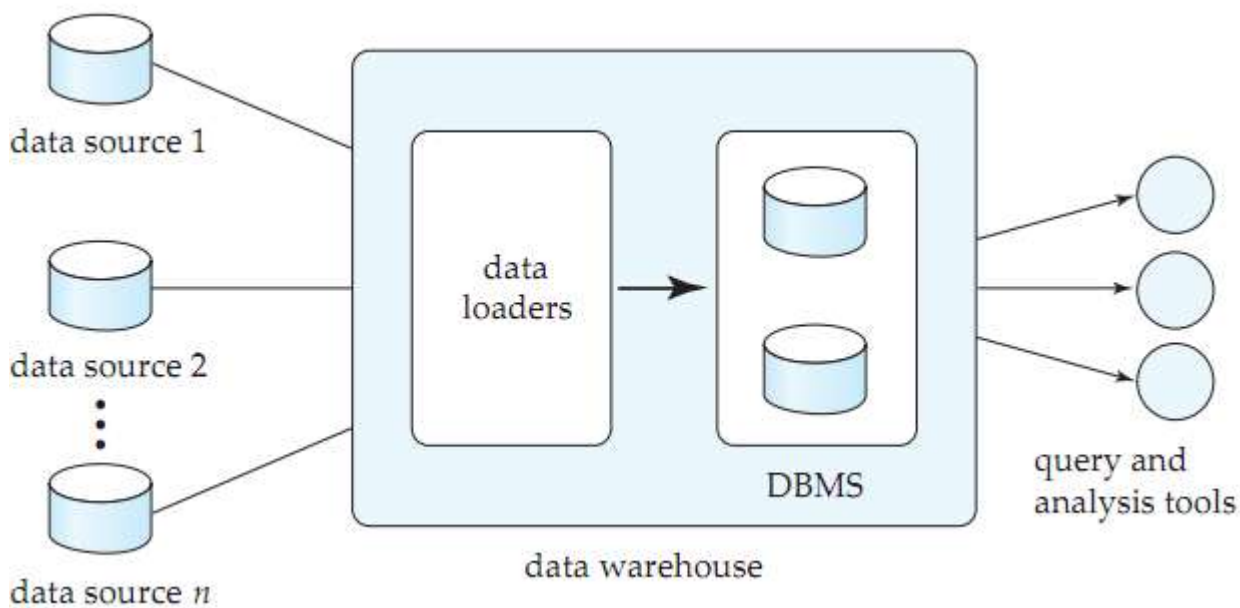
6.3.Маълумотларни сақлашни ташкиллаштириш³⁷

Катта муассасаларда жуда кўп жойларда боғлиқ маълумотлар бор ва уларни генерация қилиш учун катта миқдордаги маълумот керак. Масалан, катта маҳсулотларни тарқатишш компанияларини олсак. Уларни минглаб тармоқлари бирлаштириб турибди. Шунингдек катта муассаса мураккаб тузилишга эга эканли маълум, ва унинг бошқа тармоқлари ҳам бир биридан фарқловчи тузилишга эга.

Маълумотларни сақлаш элементлари

6.1 – расмда типик маълумотларни сақлаш архетектураси келтирилган. Шунингдек маълумотларни йиғиш, сақлаш шакллари ва маълумотларни, сўровларни таҳлиллаш келтирилган.

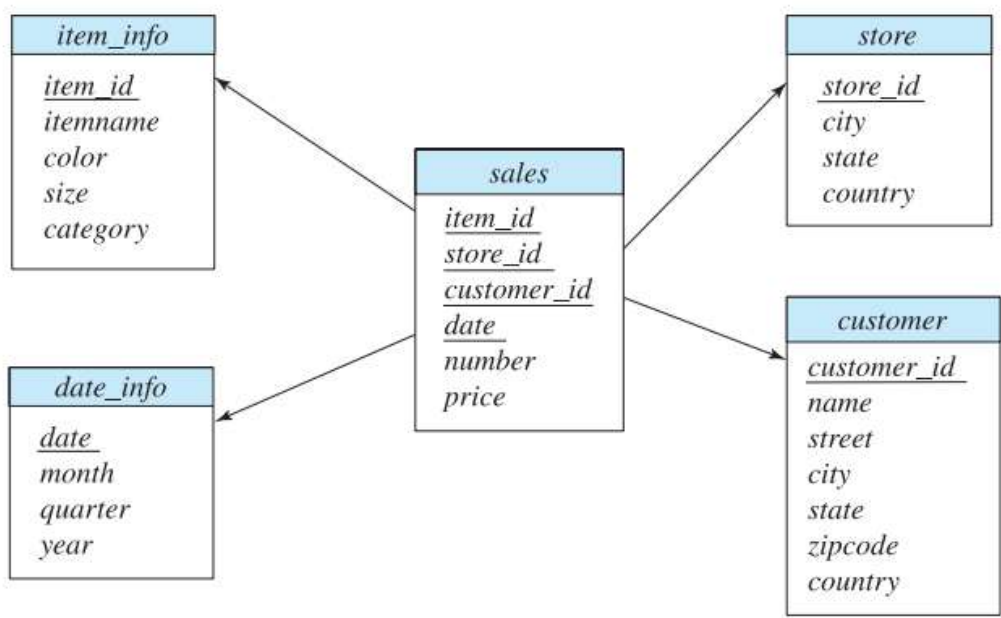
³⁷ Silberschatz, Abraham. Database system concepts // pages-889-890



6.1-расм.Маълумотларни сақлаш архитектураси.

6.4.Қоидалар омбори³⁸

Маълумотларни йиғиш тизим одатда маълумотларни таҳлил қилиш имкониятини берувчи инструментал қулайликларга эга бўлиши керак (масалан, OLAP). Шундай қилиб маълумотлар, кўп структурали ва кўп ўлчовли, ҳар хил атрибутлидир. Кўп ўлчамли маълумотларни сақловчи жадваллар фактлар жадвали деб номланади.

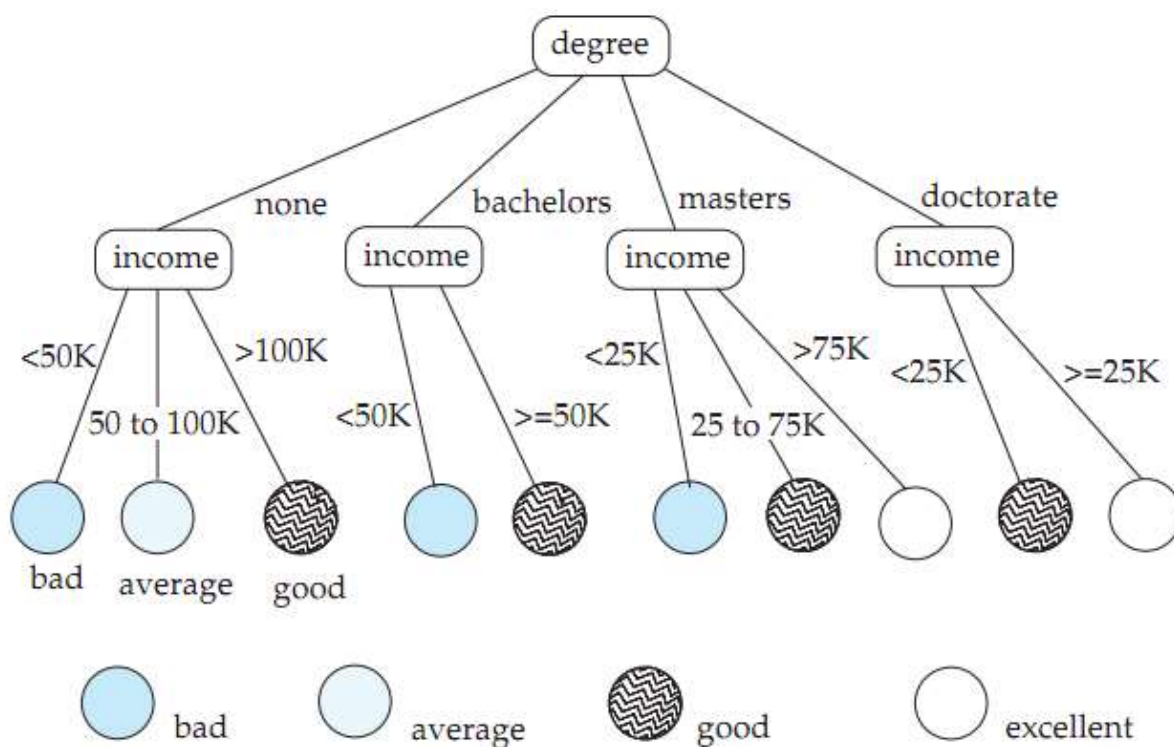


6.2-расм.Маълумотлар омборида маълумотларни сақлаш схемаси.

³⁸ 1. Silberschatz, Abraham. Database system concepts // pages-591-592

Маълумотларни сақлаш омборлари асосий рол қарор қабул қилиш қоидаларига усулрига боғлиқ равишда ўрганилади. Уларни ташкил қилиш учун одатда қуйида келтириладиган data mining шажарали қарор қабул қилиш моделидан фойдаланилади.

Ҳар бир тугун учун агар ҳамма, ёки “амалий ҳамма” ўқув тасодифлари шу синфга таалуқлилари билан боғлиқ бўлса, шунда бу синф рўйхат тугуни бўлади ва шу синф билан боғланган бўлади. Қарама қарши ҳолатда атрибутлар режалаштирилган ва шартлар асосида яратиш учун танланган шўъба (дочерние) тугун бўлиши керак. Ҳар бир шўъба тугун билан боғлиқ бўлган маълумот ўқув нусхаларининг тўплами бўла олади, бу шўъба тугуннинг режалаштирилган шартларини қаноатлантирса.



6.3-расм Класификациялаш шажараси.

6.5. Ечимлар шажарасини қуришни классификациялаш³⁹

Бу ерда қарор қабул қилиш учун ечимларнинг шажарали қуриш синфларини ажратиш масаласи қаралади. Бу масалани ечиш учун қарам алгоритмлардан фойдаланиш мумкин. Бу алгоритмлар рекурсив алгоритмлар бўлиб, шажаранинг асосий бошланғич тугунидан то энг сўнги тугунига олиб

³⁹ Silberschatz, Abraham. Database system concepts // pages-895-896

борилади. Аввалиги битта асосий бошланғич бош нукта бўлади ва барча урганиш манбалари у билан боғланган бўлади.

Ҳар бир тугун учун агар ҳамма, ёки “амалий ҳамма” ўқув тасодифлари шу синфга таалуқлилари билан боғлиқ бўлса, шунда бу синф рўйхат тугуни бўлади ва шу синф билан боғланган бўлади. Қарама-қарши ҳолатда атрибутлар режалаштирилган ва шартлар асосида яратиш учун танланган шуъба (дочерние) тугун бўлиши керак. Ҳар бир шуъба тугун билан боғлиқ бўлган маълумот ўқув усхаларининг тўплами бўла олади қайсики бу шуъба тугуннинг режалаштирилган шартларини қаноатлантирса. Юқорида келтирилган бизнинг мисолини таҳлил қилинг.

Энг яхши бўлинишлар⁴⁰

Фараз қиламиз барча ўқув танланмалари бир синфга тегишли бўлсин. Аниқ бир атрибутни танлаш ва уни тугунларга бўлишда биз унинг боғлиқчаслигини текшираамиз ва шу асоисда бўлинади. Энг юқори кўрсаткичли боғлиқмаслиқ атрибут ва шартлари танланилади.

Боғланмаслиқ тўплами S ўқув талланмалари билан берилган бўлсин. S синфнинг синфлари ва нусхалари бўлсин.

Боғланмаслиқни ҳисоблаш формулаларидан бири $Gini$ ўлчови бўлиб ҳисобланади:

$$Gini(S) = 1 - \sum_{i=1}^k p_i^2$$

Қачонки барча элементлар бир синфга тегишли бўлса, $Gini$ ўлчови 0 га тенг, Агар ҳар бир синф бир хил нусхаларга эга бўлса, у ўзининг максимумига эришишди (1 дан $1 / K$).

Боғланмаслиқни ҳисоблаш формулаларидан бири $Entropy$ ўлчови бўлиб ҳисобланади:

$$Entropy(S) = - \sum_{i=1}^k p_i \log_2 p_i$$

Бу 0 га тенг бўлади агар нусхаоларнинг барчаси бир синфга тегишли бўлса, ва ўзининг максимум қийматига эришади агар барча нусхалар бир хил элементга эга бўлса. Бу содалаштириш учун S тўплами бир нечта кичик тўпламларга бўламиз. $S_i, i = 1, 2, \dots, r$. Биз келтирилган наборни ўзгартирамиз.

$$Purity(S_1, S_2, \dots, S_n) = \sum_{i=1}^r \frac{|S_i|}{|S|} purity(S_i)$$

⁴⁰ 1. Silberschatz, Abraham. Database system concepts // pages-897-898

6.6. Маълумотларни излаш усуллари

Бунинг учун МИТларида долзарбликни ҳисоблаш, яъни ҳар бир сўров учун аниқланган маълумотлар тўпламида долзарбликни ҳисобловчи автомат (scorer, кейинги ўринларда ҳисобловчи) билан таъминланишлари лозим. Ҳисобловчилар долзарбликни сўров - маълумот жуфтлиги муносабатида шакллантирилади (ҳисоблашади). Бунда учта ғояни илгари суриш мумкин.

1. Параметрли ва зонали индексларни қуриш. Бундан иккита мақсадда фойдаланиш мумкин, яъни – улар индекслаш ва мета маълумот орқали маълумотни излаш имкониятини таъминлайди, - маълумотларни сўровга нисбатан долзарблик даражасини аниқлаш имкониятини таъминлайди.

2. Маълумотдаги муҳим терминларнинг вазнини, термин такрорланиши ҳақидаги статистика асосида таъминлаш.

3. Ҳар бир маълумотни вектор сифатида қаралса, ҳар бир маълумот учун сўровларнинг мослигини ҳисоблаш мумкин. Бу ёндашув эса векторли тўплам моделига асосланган долзарблик деб номланади.

Параметрли ва зонали индекслаш. Маълумотларни излашнинг мантиқий излаш моделларида маълумотларни терминларнинг кетма-кетлиги сифатида олинади. Аслини олганда кўп маълумотлар қўшимча тузилишларга ҳам эга. Электрон маълумотлар одатда компьютер томонидан кодланган метамаълумот (metadata) билан биргаликда бўлади. Маълумотнинг метамаълумоти деганда, унинг аниқ характеристикалари, масалан, муаллиф, номи, яратилган санаси тушунилади. Бу метамаълумотлар метамаълумот майдонларини (fields) билдиради, уларни одатда маълумот атрибутлари деб юритишади. Бу атрибутларнинг типлари ва қийматлари олдиндан чегаралаб қўйилган бўлади.

“Пўлат Мўминнинг 1950 йилдаги севги тўғрисидаги шеърлар” сўровини кўриб чиқамиз. Бу ҳолда сўров тартибланган рўйхат ва параметрик индекс орқали қайта ишланади. Унда ҳар бир атрибут учун битта параметрик индекс мавжуд бўлади. Параметрик индекс айнан мавжудларини аниқлаш имкониятини беради. Масалан, сўровдаги вақт, фақатгина параметрик индексда вақт атрибути мавжуд бўлган маълумотларни олади. Бундай сўровларни шакллантириш, яъни параметрик излашни амалга ошириш учун мулоқот ойнасидан фойдаланилади (1-расм). Бундай параметрик индексларни излаш учун В-шажара [1] каби усуллардан фойдаланиш мумкин.

Зона майдони ихтиёрий матнларни сақлаш учун мўлжалланган бўлиб одатда маълумотнинг сарлавҳаси ёки аннотациясини сақлашга мўлжалланган бўлади.

Сўровни қайта ишлаш усулини қўллаб қувватлаш учун, сарлавҳа, муаллиф, аннотация бўйича тартибланган рўйхатни ташкиллаштириш мумкин (1-жадвал).

1-жадвал.

Элементар зонали индекслаш.

Аннотация.термин → 1 → 2 → 3 → 6 → 45 → 89 → 600
Сарлавҳа.термин → 5 → 6 → 8 → 45 → 89 → 500
Муаллиф.термин → 6 → 9 → 10 → 15 → 56→500

Кодлаш орқали зонали индексларнинг ихчамлаштириш мумкин. Масалан,

Термин → 2.сарлавҳа,2.аннотация →3.Муаллиф→5.аннотация

Долзарбликнинг вазли зоналари. Параметрли ва зонали излашда мантиқий сўровлар асосида излаш усулларидадан фойдаланишдан ташқари зона ва атрибут орқали излашнинг иккинчи вариантыни кўриб чиқамиз.

Бунинг учун мантиқий сўровни q билан, маълумотни эса - d билан белгилаймиз. Долзарбликнинг вазли зона усулида (q, d) жуфтлик $[0,1]$ ораликда акслантирилади. Бунда маълумотлар зонасини чизиқли ҳисоблаш комбинациясидан фойдаланилади. Маълумотлар зонаси мантиқий қийматларни қабул қилади.

l зонали маълумотга эга маълумотлар тўпламини қараймиз. Унда $g_1, g_2, \dots, g_n \in [0,1]$ бўлсин ва

$$\sum_{i=1}^l q_i = 1 \tag{1}$$

ни қаноатлантисин. $1 \leq i \leq l$ да q сўров ва i -чи зонанинг муносабати s_i – мантиқий катталиқ бўлсин. Агар сўровнинг барча терминлари битта зонага тегишли бўлса, 1 га тенг акс ҳолда 0 га тенг. Ҳақиқатан ҳам буни $\{0,1\}$ тўпланининг зонасида терминларга асосланган ихтиёрий мантиқий функция қаноатлантириши мумкин. Бундай вазли зонани

$$\sum_{i=1}^l g_i s \tag{2}$$

каби аниқлаб олиш мумкин. Долзарбликнинг вазли зоналарини баъзан долзарблик асосида мантиқий излаш деб ҳам юритилади.

Вазли зонанинг долзарблиги қандай ҳисобланади? - деган савол пайдо бўлади. Бунинг учун ҳар бир маълумотнинг долзарблигини кетма-кет ҳисоблаш орқали ҳар хил зоналарни йиғиндиси кўринишида амалга ошириш мумкин. Тартибланган индекс орқали ҳам вазли зонанинг долзарблигини ҳисоблаш мумкин. Фараз қиламиз, q_1 ва q_2 терминлардан иборат ва

мантикий амали билан боғланган q сўров берилган бўлсин. Агар иккила термин ҳам бир зонага тегишли бўлса, 1 акс ҳолда 0 (1-алгоритмга қаранг). Алгоритмни мураккаб сўровлар ва мантикий функциялар учун кенгайтирилган вариантларини ҳам ёзиш мумкин.

1-алгоритм. Вазли зонанинг долзарблиги.

```
ZoneScore( $q_1, q_2$ ){
  double scores[N] = isEmpty;
  Constant  $g[l]$ ;
   $p_1 = postings(q_1)$ ;
   $p_2 = postings(q_2)$ ;
  while ( $p_1 \neq \text{Null}$  and  $p_2 \neq \text{Null}$ ) do
    if ( $\text{docID}(p_1) == \text{docID}(p_2)$ ){
       $\text{score}[\text{docID}(p_1)] = \text{WeightedZone}(p_1, p_2, g)$ ;
       $p_1 = \text{next}(p_1)$ ;
       $p_2 = \text{next}(p_2)$ ; }
    else
      if ( $\text{docID}(p_1) < \text{docID}(p_2)$ )  $p_1 = \text{next}(p_1)$ ;
      else
        if ( $\text{docID}(p_1) > \text{docID}(p_2)$ )  $p_2 = \text{next}(p_2)$ ;
  return scores
}
```

Қийматни машина ўқуви⁴¹ ёрдамида аниқлаш.

Вазли зоналарнинг долзарблигида g_i ларнинг қийматини қандай аниқланади?-деган савол пайдо бўлади. Бунда экспертлар (фойдаланувчилар) орқали аниқланиши мумкин. Лекин, жуда кўп ҳолларда қийматлар олдиндан баҳоланган, ўқув мисоллари асосида аниқланади. Бу усул маълумотларни излашдаги машина ўқувида асосланган долзарблик усуллари умумий синфига таалуқлидир. Долзарбликнинг вазли зоналилари учун қисқа келтириб ўтамиз.

1. Уқув тажрибаларига асосланган мисоллар тўплами мавжуд бўлсинки, унда ҳар бир элемент аниқ ифодадан ташкил топган. Аниқ ифода эса, q сўров, d маълумот ҳамда q ва d лар учун долзарблик қийматини белгиласин. Энг содда вариантда долзарбликни бинарли деб қараш мумкин. Масалан, долзарб – 1 ёки долзарбмас – 0. Мураккаб методологияларини қўллаш учун махсус нозик баҳолаш усуллари қўлланилади.

⁴¹ Машина ўқуви – “Машинная обучения” рус тилида, “Computer Learn” инглиз тилида, компьютерда олдин аниқлаган амаллар (билимлар) ёрдамида жараёнини амалга ошириш.

2. g_i қийматлар уқув асосида аниқланади, чунки олинаётган қиймат долзарблик қийматини аппроксимациялаган бўлиши керак.

Долзарбликнинг вазли зоналари учун ушбу жараённи сараланган коэффициентли чизиқли функция сифатида қараш мумкин. Коэффициентлар эса зонага тегишли мантиқий қийматлар. Бу методологиянинг машаққатли томони шундаки, уқув асосида долзарбликнинг қийматини йиғишдир, айниқса, маълумотлар тез-тез ўзгариб турса (вебга ўхшаш). Оддий оптимизация масаласига доир машина уқуви асосида g_i ларнинг қийматини аниқловчи масала келтирамиз.

Долзарбликнинг вазли зонаси учун оддий масала қараймиз, унда ҳар бир маълумот сарлавҳа (title) ва аннотация (body) зоналарига эга бўлсин. Берилган q сўров ва d маълумот учун $s_T(d, q)$ ва $s_B(d, q)$ мантиқий функцияларни аниқлаймиз. Қайсики, мантиқий функциялар q сўров d маълумотнинг сарлавҳаси (аннотацияси) бўлиши ёки бўлмаслигини аниқласин. Бунда сўров терминлар учун ва мантиқий амалдан фойдаланиш мумкин. Бир ва нол оралиғидаги долзарблик қийматини ҳар бир (маълумот-сўров) жуфтлиги асосида ҳисоблаш учун $s_T(d, q)$ ва $s_B(d, q)$ ларнинг қийматларидан ва $g \in [0,1]$ ўзгармасдан фойдаланиш мумкин.

$$Result(d, q) = g \cdot s_T(d, q) + (1 - g) \cdot s_B(d, q) \quad (3)$$

$\Phi_j = (d_j, q_j, r(d_i, q_j))$ учлик асосида аниқланган уқув мисоли асосида g ўзгармасни аниқлашни келтирамиз. Ҳар бир ўқув мисоли d_i маълумот, q_i сўров ва $r(d_i, q_j)$ долзарблик қиймати асосида аниқланади (2-жадвал).

2-жадвал.

Уқув мисоллари.

Уқув мисоли	DocI D	сўров	s_T	s_B	Долзарблик қиймати
Φ_1	1	Тинчлик	1	1	1
Φ_2	28	Компьютер	0	1	0
Φ_3	80	Китоб	0	1	1
Φ_4	231	Маълумот	1	0	0
Φ_5	344	Ахборот	1	0	1
Φ_6	1991	Излаш	0	1	1
Φ_7	1177	долзарб	0	1	0

Ҳар бир Φ_1 уқув мисоли учун $s_T(d, q)$ ва $s_B(d, q)$ мантиқий қийматлар мавжуд. Шунинг учун долзарбликни ҳисоблаш учун юқоридаги (3) формуладан фойдаланамиз.

Инсон фактори орқали келтирилган долзарблик қиймати ва (3) долзарблик қийматларини таққослаймиз. Қиймат хатолиги функциясини қуйидагича аниқлаймиз.

$$\varepsilon(g, \Phi_j) = (r(d_j, q_j) - \text{Result}(d_j, q_j))^2 \quad (4)$$

Бунда инсон фактори 0 ёки 1 билан ифодаланади. Энди бу хатоларни йиғамиз ва ўқув мисоллари учун қуйидагига эга бўламиз.

$$\sum_j \varepsilon(g, \Phi_j) \quad (5)$$

Энди ўқув мисоллари асосида g ўзгармасни аниқлаш мумкин. Бунинг учун (5) ифодани энг кичик бўлган қийматларидаги g ни аниқлаш керак.

Ҳақиқатан ҳам (5) ифода ёрдамида g нинг энг яхши қийматларини аниқлаш мумкин. Буни эса $[0,1]$ оралиғида g га боғлиқ квадрат функция орқали минимизациялаш масаласига келади.

g нинг энг самарали қиймати. Ихтиёрий Φ_j учун $s_T(d_j, q_j) = 0$ ва $s_B(d_j, q_j) = 1$ бўлсин. Долзарблик (3) ифода орқали ҳисобланса 1- g га тенг бўлади. (3) ифодани $s_T(d_j, q_j)$ ва $s_B(d_j, q_j)$ ларнинг бошқа комбинациялари учун ҳам ёзиш мумкин(3-жадвал).

3-жадвал.

s_T ва s_B ларнинг 4 та комбинацияси

$s_T(d_j, q_j)$	$s_B(d_j, q_j)$	Долзарблик
0	0	0
0	1	1- g
1	0	g
1	1	1

$s_T(d_j, q_j) = 0$ ва $s_B(d_j, q_j) = 1$ бўлганда инсон фактори долзарб $\{1\}$ (мос равишда, долзарб эмас $\{0\}$) ўқув мисолларини сонини n_{01r} (мос равишда n_{01n}) билан белгилаймиз. Шунда (5) ифодани $s_T(d_j, q_j) = 0$ ва $s_B(d_j, q_j) = 1$ учун қуйидагича ёзиш мумкин.

$$[1 - (1 - g)]^2 n_{01r} + [0 - (1 - g)]^2 n_{01n}$$

Худди шундай $s_T(d_j, q_j)$ ва $s_B(d_j, q_j)$ ларнинг бошқа комбинациялар учун (5) ифода орқали аниқлаган йиғиндини ёзиш мумкин.

$$(n_{01r} + n_{10n})g^2 + (n_{10r} + n_{01n})(1 - g)^2 + n_{00r} + n_{11n} \quad (6)$$

(6) ифодани g бўйича содалаштириб, нолга тенглаштирсак, g нинг энг самарали қийматини аниқлаймиз.

$$\frac{n_{10r} + n_{01n}}{n_{10r} + n_{10n} + n_{01r} + n_{01n}} \quad (7)$$

Термин частотаси ва вазни. Маълумотнинг долзарблиги унинг зоналарига боғлиқ эди. Энди бошқача ёндашамиз, яъни сўровларда кўп учрайдиган терминнинг зонаси ёки маълумоти учун сўровнинг долзарблигини инобатга оламиз ва унга катта қиймат киритамиз. Бу фараз сўровларни эркин киритиш учун қулайдир, яъни МИТларда сўровларнинг эркин, ёрдамчи амалларсиз киритиш (мантикий амаллари каби) назарда тутилган. Бу ёндашув кўпроқ веб технологияларга асосланган ва унда сўров терминларнинг тўплами сифатида қаралади. Ҳақиқатан ҳам, маълумотларнинг кўрсаткичларини ҳисоблаш учун сўровдаги терминларнинг маълумот қийматларини йиғиш кифоядир.

Ушбу терминнинг ҳар бир маълумот қийматини бирлаштириш, яъни маълумотдаги терминнинг учрашиш сонига боғлиқ бўлиб, терминнинг вазнини аниқлашга олиб келади. Бундан эса, d маълумотдаги t терминнинг вазни орқали q сўров ва d маълумотнинг ўзаро муносабати аниқланади. Бу вазни d маълумотдаги t терминнинг сони билан ҳам белгилаш мумкин. Бу алгоритм терминнинг частотаси (term frequency) деб юритилади ва $tf_{t,d}$ каби белгиланади, бунда индексдаги t - термин ва d - маълумотни билдиради.

Бирор d маълумот учун D вазнлар тўплами (ихтиёрий бирор бир функция, ёки d маълумотдаги t терминнинг сони орқали аниқланган ва ҳар доим мусбат, бутун сонлар) ни d маълумотнинг ифодасини сонлар кўринишида изоҳлаши мумкин. Бундай ёндашув илмий журналларда термин копи (bag of words model) деб юритилади. Бу модел асосида ҳар бир терминнинг маълумотда учрашиш сони маълумотдаги терминларнинг тартиби билан изоҳланади. Бунда фақат ҳар бир терминнинг учрашиш сони ҳақидаги маълумотдан фойдаланамиз. Шундай қилиб, “Алининг укаси Вали” ва “Валининг укаси Али” маълумотлари бир хил қийматлидир, яъни терминлар копи бир хил. Берилган маълумотдаги барча терминларнинг вазнлари бир хилми деган саволга жавоб излаймиз. Ҳақиқатан ҳам маълумотдаги терминларнинг вазнлари бир хилмас. Чунки унда стоп-термин (stop words)лар, индекслаштиришга инобатга олинмайдиган терминлар, излаш ва долзарбликка таъсир этмайдиган терминлар мавжуд.

Тескари маълумотли частота. Одатда сўровдаги терминларнинг барчаси муҳим ўрин тутади деб қаралади. Юқорида келтирилганлар асосида бундай ёндашувда жуда кўп камчилик мавжудлигини кўришимиз мумкин. Аммо, асли олганимизда сўровдаги терминларнинг аҳамияти турлича, яъни кам ва баъзан нол қийматли терминлар ҳам учрайди. Масалан, автомобил саноати учун маълумотлар тўплами мавжуд бўлсин. Бу тўпланинг деярли барча маълумотларида “авто” терминини учратиш мумкин. Ушбу

камчиликни бартараф этиш учун, кам таъсирчан термин тушунчасини киритамизки, маълумотда тез-тез учраб турувчи ва маълумотнинг долзарблигига таъсир кўрсатмайдиган термин назарда тутилади. Тўпلامда катта частотали терминларнинг вазнини камайтириш ғояси келади. Бу ғоянинг асоси шундаки, tf терминнинг вазнини коэффиценти орқали кичиклаштиришдир, яъни унинг тўпلامдаги частотасининг катталашини ҳисобига.

Бу ғоянинг ўрнига жуда кўп ҳолларда df_i маълумот частотаси (document frequency) тушунчаси ишлатиладики, унда t терминдан иборат бўлган df_i тўпلامдаги маълумотлар орқали ифодаланади. Бу сўров долзарблигига нисбатан маълумотлар орасидаги фарқни топиш орқали изоҳланади. Аммо, маълумотларнинг ўз статик кўрсаткичлари орқали фойдаланиш самаралироқдир. Масалан, берилган терминдан мос маълумотлар сони. Тўпلامдаги термин частотаси ва маълумотли частота 4-жадвалда келтирилган. Бунда cf тўпلامдаги термин частотаси ва df маълумот частотаси турлича бўлиши мумкин. Жадвалда хусусий ҳол учун терминларнинг частотаси бир бирига яқин, аммо маълумот частоталари анча фарқли. Бундан эса, “биринчи” терминини сақловчи маълумотлар шу терминли сўровлар муносабатига қараганда “иккинчи” терминини сақловчи маълумотларнинг шу терминли сўровлар муносабати катта аҳамиятга эга.

4-жадвал.

Тўпلامдаги частота ва маълумотдаги частота.

Термин	cf	df
Биринчи	128	111980
Иккинчи	125	772005

Маълумот частотасини уни вазнини яхшилаш учун қандай фойдаланиш мумкин. Тўпلامдаги маълумотлар сонини N билан белгилаймиз ва t терминнинг тескари маълумотли частотани қуйидагича аниқлаймиз.

$$idf_t = \log \frac{N}{df_t} \quad (8)$$

Шундай қилиб, кўп учрайдиган терминларга нисбатан кам учрайдиган терминларнинг тескари маълумот частотаси катта бўлиб ҳисобланади (5-жадвал). Бунда тўпلامдаги маълумотлар сони $N = 32021177$ ва (8) ифода билан ҳисобланган.

Тескари маълумот частотасига мисол

Термин	cf_t	idf_t
Бир	1980	4.209
Икки	1981	4.21
Уч	2004	4.20355
Тўрт	2005	4.2033

Терминнинг тескари маълумот частотаси ва частоталар комбинациялари асосидаги вазн. Ҳар бир d маълумотда ҳар бир t терминнинг вазнини аниқлаш учун d маълумотда t терминнинг комбинациялашган частотаси tf (term frequency) ва тескари маълумот частотаси idf (inverse document frequency). $tf-idf$ вазнини ҳар бир t терминга мос d маълумот вазнини бирлаштириш асосида аниқлаймиз.

$$tf - idf_{t,d} = tf_{t,d} \times idf_{t,d} \quad (9)$$

Бошқача қилиб айтганда, d маълумотдаги t терминнинг $tf - idf_{t,d}$ вазни қуйидаги хусусиятларга эга.

1. Агар катта сонли бўлмаган маълумотларда t термин кўп марта учраса, у максимал қийматга эга бўлади (бу шартнинг ўзи бошқа маълумотлардан фарқланишини билдиради).

2. Агар қайсидир маълумотда фақатгина бир неча марта учраса ёки кўп маълумотларда учраса, у кичиклашади.

3. Агар термин барча маълумотларда учраса, у минимал қийматга эга бўлади.

Ҳар қандай маълумотни вектор сифатида шакллантириш мумкин, унда ҳар бир элемент учун мос луғатдаги термин ва вазн (10) ифода билан ҳисоблаш мумкин.

$$Result(q, d) = \sum_{t \in d} tf - idf_{t,d} \quad (10)$$

Маълумотда учрамайдиган луғатдаги терминларнинг вазни 0 га тенг. Бу векторли ифодалаш вазн ва долзарблик учун муҳимдир. D маълумотнинг аҳамиятлилиги шундаки, сўровнинг барча терминлари жорий маълумотга тегишлилигидадир. Аммо, сўровдаги ҳар бир терминни маълумотга тегишлилигини йиғиш эмас балки уларнинг $tf-idf$ вазнини йиғиш керак.

Назорат саволлари:

1. Маълумотларни жамлаш ва излаш нима учун керак?
2. Маълумотларни йиғиш тизими ҳақида гапириб беринг?

3. Маълумотларни сақлашни ташкиллаштириш чора тадбири?
4. Қоидалар омбори қандай яратилади?
5. Класификациялаш шажарасига мисол келтиринг?
6. Параметрли ва зонали индекслаш ниму учун керак?
7. Қийматни машина ўқуви⁴² ёрдамида аниқлаш усулини тушунтириб беринг?

Фойдаланилган адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. см. 2011. ISBN 978-0-07-352332-3.
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)"(2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000).
5. <http://www.colapublib.org/periodicals/>
6. <http://houstonlibrary.org/research/databases>

⁴² Машина уқуви – “Машинная обучения” рус тилида, “Computer Learn” инглиз тилида, компьютерда олдин аниқлаган амаллар (билимлар) ёрдамида жараёнини амалга ошириш.

IV БЎЛИМ

АМАЛИЙ МАШҒУЛОТ
МАТЕРИАЛЛАРИ

IV. АМАЛИЙ МАШҒУЛОТ МАТЕРИАЛЛАРИ

1-амалий машғулот (4 соат). SQL тилида маълумотлар базасини яратиш

Ишдан мақсад: SQL тилида Университет маълумотлар базасини ташкил этиш учун зарур бўлган маълумотларни йиғиш ва уларнинг реляцион моделини ташкил этиш. SQL тили МВ яратиш имкониятларини ўрганиш.

Калит сўзлар: Data Abstraction, Database Languages, Relational Databases, A sample E-R diagram. SQL, database, university database Database-management system, DBMS, Database-system applications, File-processing systems, Data inconsistency, Consistency constraints, Data abstraction

Масаланинг қўйилиши: Университет маълумотлар базасини реляцион моделини яратиш.

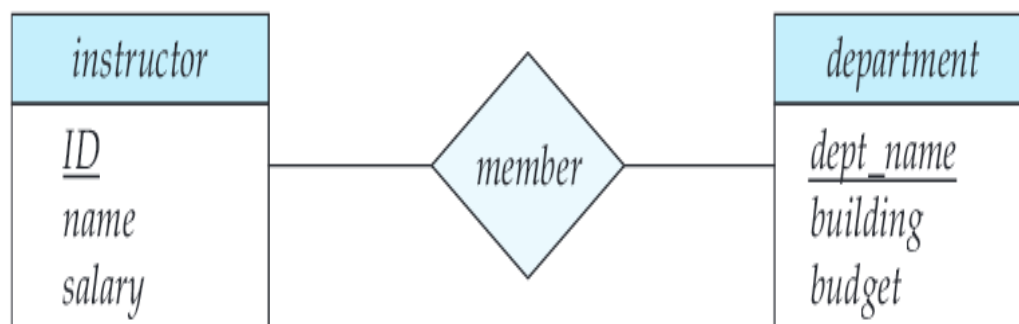
Ушбу амалий иш давомида қуйидагиларни **бажариш лозим:**

- MS SQL Serverни ўрнатиш
- Маълумотлар базасини лойиҳалаш
- Жадвалларни яратиш
- Керакли маълумот билан тўлдириш.

Ишни бажариш учун намуна

Амалий қисм

Малумки, Университет маълумотлар базасини ташкил этишда реляцион модель яратиш талаб этилади. Бунинг учун биринчи навбатда E-R диаграмма қуриш зарур.



1-расм. E-R диаграмма

Ушбу диаграмма асосида SQL МВБТ ни шакллантириш учун қуйидаги таблицалар курилади.

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

2-расм. Релизацион малумотлар базаси

Агарда МВда иштирок этадиган жадваллар бир-бири билан бўтланган бўлса, бундай МВни реляцион МВ деб аташ қабул қилинган. Бунда жадвалларни бир-бири билан боғлаш учун умумий хусусиятга эга бўлган уникал майдон тушунчаси киритилган. Ушбу тушунча баъзан МВнинг калитли майдони деб ҳам аталади. Жадвалнинг бир-бири билан боғланиш структураси боғланиш схемаси дейилади. МВ доимо ўзгариб туради: унга янги ёзувлар, борларига эса янги элементлар қўшилади. Реляцион маълумотлар базаси қуйидаги параметрлар билан баҳоланади:

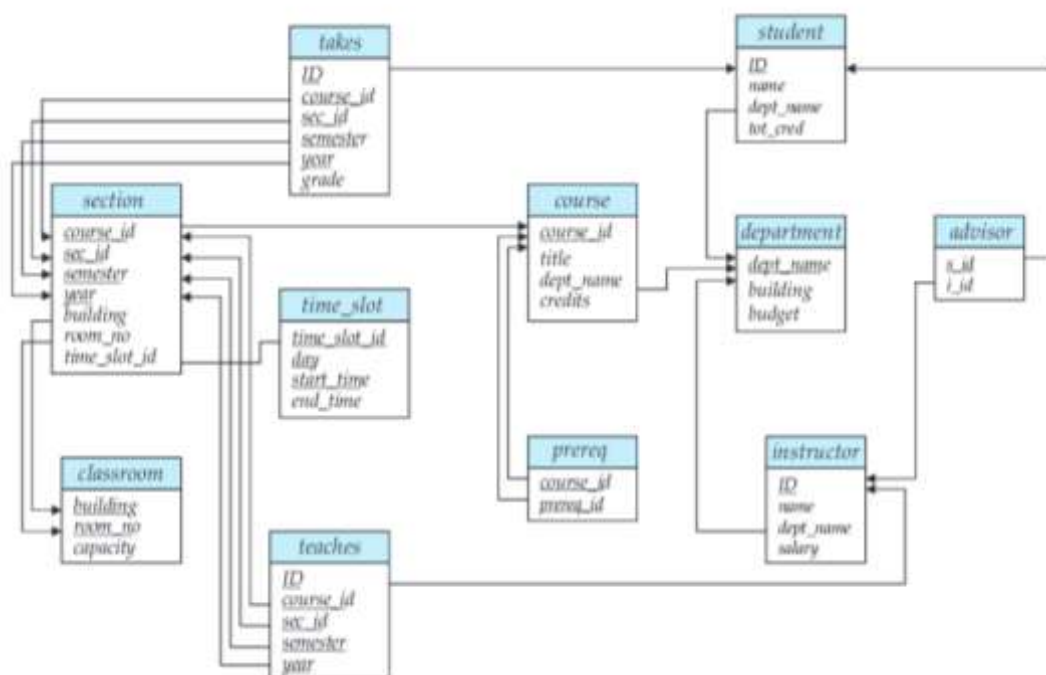
1. Простота-Соддалик
2. Гибкост-Мосланувчанлик
3. Точност- Аниқлилик математик аниқ усуллар билан МВ манипуляция қилинади,
4. Секретност-Махфийлик
5. Связанност-Боғлиқлик

6. Независимост-Маълумотлар боғлиқлиги йўқлиги,

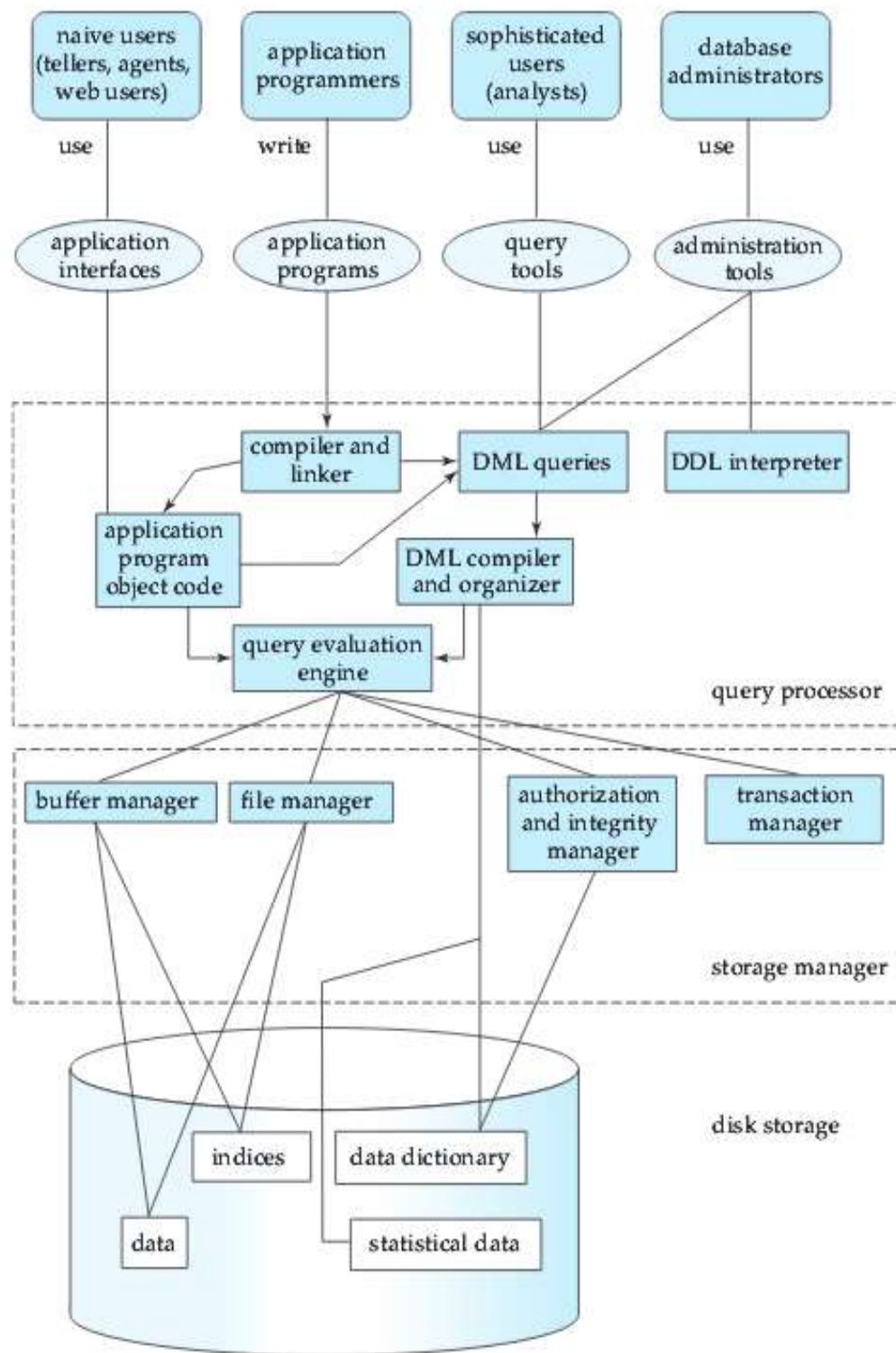
7. Маълумотлар билан манипуляция қилиш тили.

Баъзан, МВ ишлатилиши самарадорлигини ошириш мақсадида унинг тузилиши ҳам ўзгартирилиб турилади. Бу ҳолда МВнинг иэрархик ва тармоқли моделлари вужудга келади. МВни ташкил қилиш, уни тўлдириш, нухасини олиш каби вазифаларни бажариш учун махсус программа таъминоти бўлиш лозим. Бундай программа таъминоти МВБС (юқорида қайд қилганимиздек) дейилади.

Релизацион малумотлар базасидан фойдаланиб, университетнинг умумий маълумотлар базасини қуриш ва тизим аппарат таъминотига қўйилган талаб SQL да қуйидагича шакллантирилади.



3-расм. Университет маълумотлар базаси умумий схемаси



4-расм. Система архитектураси

Ушбу диаграмма асосида SQL МВБТ ни шакллантириш учун қуйидаги жадваллар қурилади.

ID	name	salary	dept_name	building	budget
22222	Einstein	95000	Physics	Watson	70000
12121	Wu	90000	Finance	Painter	120000
32343	El Said	60000	History	Painter	50000
45565	Katz	75000	Comp. Sci.	Taylor	100000
98345	Kim	80000	Elec. Eng.	Taylor	85000
76766	Crick	72000	Biology	Watson	90000
10101	Srinivasan	65000	Comp. Sci.	Taylor	100000
58583	Califieri	62000	History	Painter	50000
83821	Brandt	92000	Comp. Sci.	Taylor	100000
15151	Mozart	40000	Music	Packard	80000
33456	Gold	87000	Physics	Watson	70000
76543	Singh	80000	Finance	Painter	120000

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

course_id	title	dept_name	credits
BIO-101	Intro. to Biology	Biology	4
BIO-301	Genetics	Biology	4
BIO-399	Computational Biology	Biology	3
CS-101	Intro. to Computer Science	Comp. Sci.	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3
CS-319	Image Processing	Comp. Sci.	3
CS-347	Database System Concepts	Comp. Sci.	3
EE-181	Intro. to Digital Systems	Elec. Eng.	3
FIN-201	Investment Banking	Finance	3
HIS-351	World History	History	3
MU-199	Music Video Production	Music	3
PHY-101	Physical Principles	Physics	4

course_id	prereq_id
BIO-301	BIO-101
BIO-399	BIO-101
CS-190	CS-101
CS-315	CS-101
CS-319	CS-101
CS-347	CS-101
EE-181	PHY-101

Figure 2.2 The *course* relation.

Figure 2.3 The *prereq* relation.

course_id	sec_id	semester	year	building	room_number	time_slot_id
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

Figure 2.6 The *section* relation.

ID	course_id	sec_id	semester	year
10101	CS-101	1	Fall	2009
10101	CS-315	1	Spring	2010
10101	CS-347	1	Fall	2009
12121	FIN-201	1	Spring	2010
15151	MU-199	1	Spring	2010
22222	PHY-101	1	Fall	2009
32343	HIS-351	1	Spring	2010
45565	CS-101	1	Spring	2010
45565	CS-319	1	Spring	2010
76766	BIO-101	1	Summer	2009
76766	BIO-301	1	Summer	2010
83821	CS-190	1	Spring	2009
83821	CS-190	2	Spring	2009
83821	CS-319	2	Spring	2010
98345	EE-181	1	Spring	2009

Figure 2.7 The *teaches* relation.

Университетни ташкилий асосига кўра, университетдаги умумий жараёни ифодалаш учун таблицларнинг ўзаро боғланиш структурасини ишлаб чиқиш учун ишда энг аввало таблицаларни сошлаш ишконияти шакллантирилади. Университет ўқув хоналари, кафедралар, дарс соатлари, ўқитувчилар, талабалар учун малумотлар базасини шакллантирувчи элементлар қуйидаги тартибда амалга оширилади.

```

classroom(building, room_number, capacity)
department(dept_name, building, budget)
course(course_id, title, dept_name, credits)
instructor(ID, name, dept_name, salary)
section(course_id, sec_id, semester, year, building, room_number, time_slot_id)
teaches(ID, course_id, sec_id, semester, year)
student(ID, name, dept_name, tot_cred)
takes(ID, course_id, sec_id, semester, year, grade)
advisor(s_ID, i_ID)
time_slot(time_slot_id, day, start_time, end_time)
prereq(course_id, prereq_id)

```

Университет маълумотлар базасининг умумий структуравий кўриниши SQL да куйидагича таснифланади.



6-расм. Университет SQL маълумотлар базаси

```

create table department
(dept_name varchar (20),
building varchar (15),
budget numeric (12,2),
primary key (dept_name));

create table course
(course_id varchar (7),
title varchar (50),
dept_name varchar (20),
credits numeric (2,0),
primary key (course_id),
foreign key (dept_name) references department);

create table instructor
(ID varchar (5),

```



```

name          varchar (20) not null,
dept_name     varchar (20),
salary        numeric (8,2),
primary key (ID),
foreign key (dept_name) references department);

```

```

create table section
(course_id     varchar (8),
sec_id        varchar (8),
semester      varchar (6),
year          numeric (4,0),
building      varchar (15),
room_number   varchar (7),
time_slot_id  varchar (4),
primary key (course_id, sec_id, semester, year),
foreign key (course_id) references course);

```

```

create table teaches
(ID           varchar (5),
course_id    varchar (8),
sec_id       varchar (8),
semester     varchar (6),
year         numeric (4,0),
primary key (ID, course_id, sec_id, semester, year),
foreign key (course_id, sec_id, semester, year) references section,
foreign key (ID) references instructor);

```

1. Университет маълумотлар базаси учун SQL сўровларнинг шаклланиши

Агар а relation дан студент ёки бошқа бўйруқларни ўчириш талаб этилса, **delete from student**; амалини бажариш кифоя ҳисобланади. SQL юқорига сўровларга жавоб бериш ёки банд этиш талаб этилса, сўровларни қуйидагича ёзиш мумкин.

```

select name, instructor.dept_name, building from instructor, department
where instructor.dept_name= department.dept_name;

```

мисол учун реализацион схемадан Cartesian product *instructor* and *teaches* жадваллар учун қуйидагича бўлади.

```

(instructor.ID, instructor.name, instructor.dept_name, instructor.salary
teaches.ID, teaches.course_id, teaches.sec_id, teaches.semester, teaches.year)

```

Мисол сифатида реализацион моделдан 2009 йил кузги ва 2010 цил баҳорги семестрларни ўқитиладиган фанларни танлаш учун SQL да қуйидагича дастур ёзилади.

```

(select course_id from section
where semester = 'Fall' and year= 2009) union
(select course_id from section
where semester = 'Spring' and year= 2010);

```

Агар реализацион моделдаги ўзгаришларни қайтадан сақлашса, яни ўзаро бир хил ўхшашлик 2009 ва 2010 ўқув йилида мавжудлиги топилса **union жойига union all** га алмаштирилади:

```
(select course id from section
where semester = 'Fall' and year= 2009) union all
(select course id from section
where semester = 'Spring' and year= 2010);
```

2009 йил кузги ва 2010 баҳорги ўқув йилида барча фанларни топиш керак бўлса қуйидагича SQL да ёзишимиз мумкин.

```
(select course id from section
where semester = 'Fall' and year= 2009) intersect
(select course id from section
where semester = 'Spring' and year= 2010);
```

2009 йил кузги семестрида мажуд ва 2010 йил баҳорги семестрда фанларнинг тафовутини кўриш керак бўлса қуйидагича SQL да ёзишимиз мумкин.

```
(select course id from section
where semester = 'Fall' and year= 2009) except
(select course id from section
where semester = 'Spring' and year= 2010);
If we want to retain duplicates, we must write except all in place of except:
(select course id from section
where semester = 'Fall' and year= 2009) except all
(select course id from section
where semester = 'Spring' and year= 2010);
```

Умумий бўлим гуруҳлари умумлаштириш сифатида, "Баҳор 2010 семестр бир фан ҳар бир бўлимида ўқитувчилар сонини топиш." Сўроғини кўриш яратиш учун семестр мавжуд бўлган курс бўлимлари teaches реализация қилинади. Шу билан бирга, уларнинг шаклланиши ҳар бир ўқитувчи бўлими номини олиш учун instructor билан боғланади. Шундай қилиб, бу сўровини ёзиш жараёни қуйидагича:

```
select dept name, count (distinct ID) as instr count from instructor natural
join teaches
where semester = 'Spring' and year = 2010 group by dept name;
select name from instructor
```

*where salary > some (select salary from instructor
where dept name = 'Biology');*

Кичик сўровлар:

*(select salary from instructor
where dept name = 'Biology')*

Назорат саволлари:

1. Маълумотлар базасини бошқарув тизими билан ишлашда SQL тилининг роли хақида тушунча беринг?
2. SQL тилининг асосий вазифаси нималардан иборат?
3. SQL тили МВ да бошқариш вазифасини бажарадими?
4. SQL тили барча МВБТ ларида ишлатиладими?
5. SQL тили ва унинг яратилиш тарихи хақида тушунча беринг.
6. SQL тилининг стандартлари ва МВБТ версиялари диалектлари хақида тушунча беринг.
7. SQL ни ишлаш схемаси хақида тушунча беринг.
8. SQL тилининг қандай функционал имкониятлари мавжуд.
9. SQL тили бажарадиган функциялари.
10. SQL тили имкониятлари
11. SQL тили нечта қисмдан иборат.
12. Маълумотларни аниқлаш операторлари ДДЛ (маълумотларни аниқлаш тили) хақида тушунча беринг.
13. Университет маълумотлар базасини яратиш умумий структураси
14. Университет маълумотлар базасини яратишда қандай SQL сўровлари ишлатилади
15. SQL да кафедра сўровларини яратинг
16. SQL да малака ошириш маълумотлар базасини яратинг сўровларини яратинг.

Адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. cm. 2011. ISBN 978-0-07-352332-3
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation (2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)" (2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000)

2-амалий машғулот (2 соат). SQL тилида ER диаграммасини яратиш.

Ишдан мақсад: Университет маълумотлар базаси учун ER диаграммалар қуриш SQL тилида Университет реализацион модели асосида маълумотлар базасини яратишнинг SQL тилида имкониятларини таҳлил қилиш.

Калит сўзлар: Data-definition language, View definition, SQL Data Definition. SQL, Data inconsistency, Consistency constraints, Data abstraction

Масаланинг қўйилиши: Университет маълумотлар базаси учун ER диаграммалар яратиш.

Ишни бажариш учун намуна

Қуйидаги соҳалар учун E-R диаграммаларни ўзаро боғланишларини SQL да ташкил.

1-қадам.



Figure 6.1 E-R diagram for a car insurance company.

2-қадам



Figure 6.2 E-R diagram for a university.

3-қадам

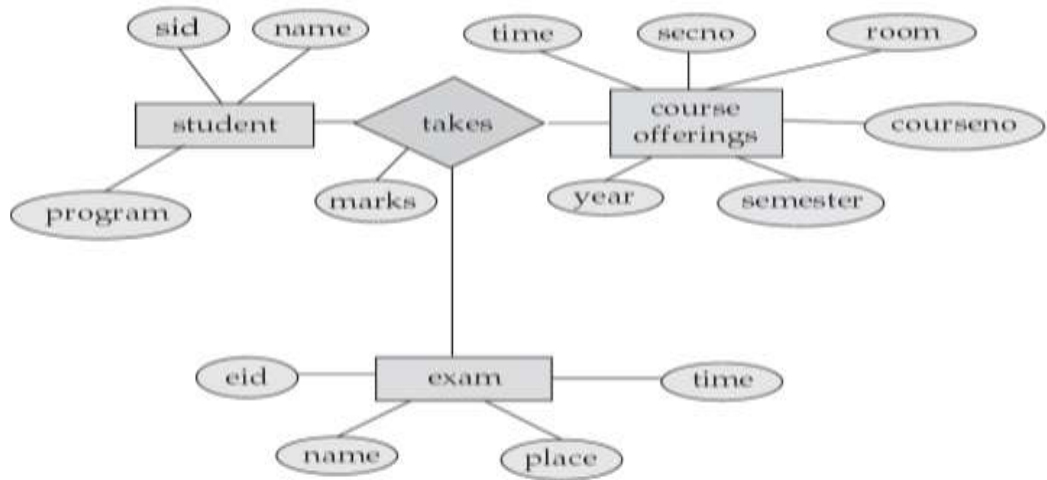


Figure 6.3 E-R diagram for marks database.

4-қадам

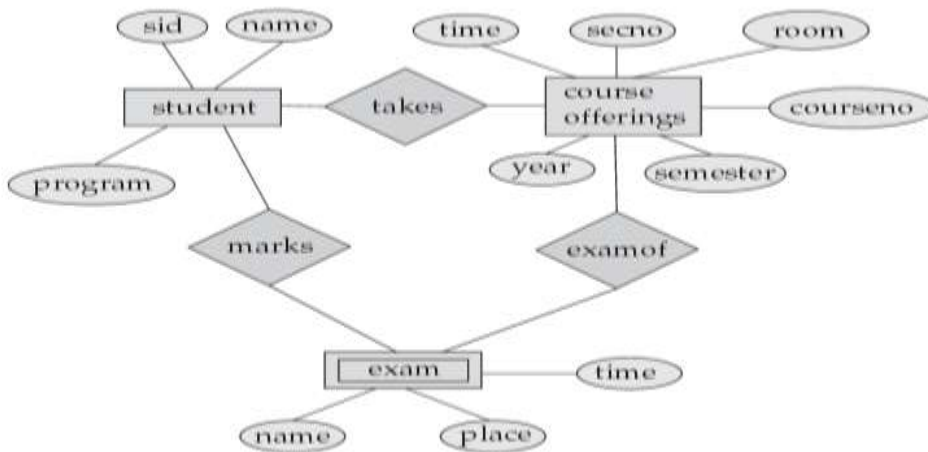


Figure 6.4 Another E-R diagram for marks database.

5-қадам.

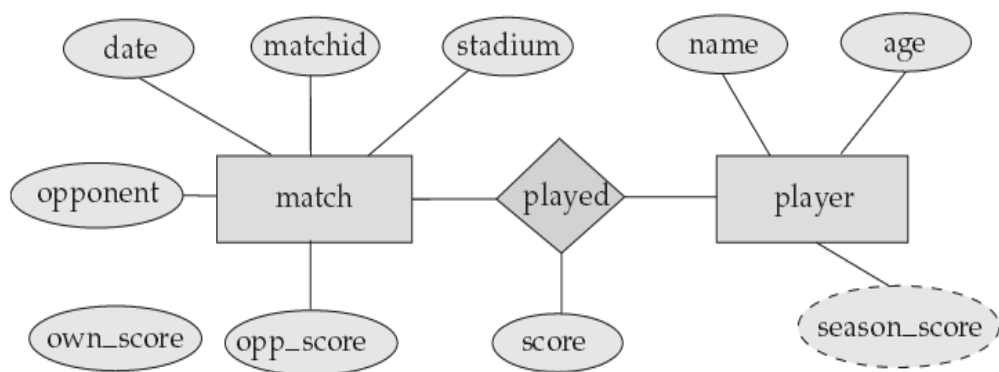


Figure 6.5 E-R diagram for favourite team statistics.

6-қадам

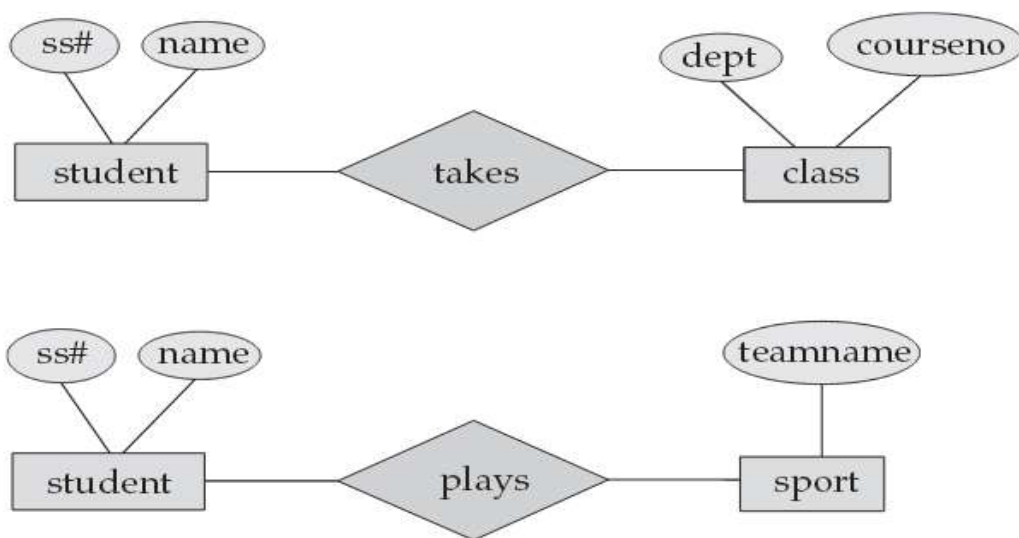


Figure 6.6 E-R diagram with entity duplication.

7-қадам

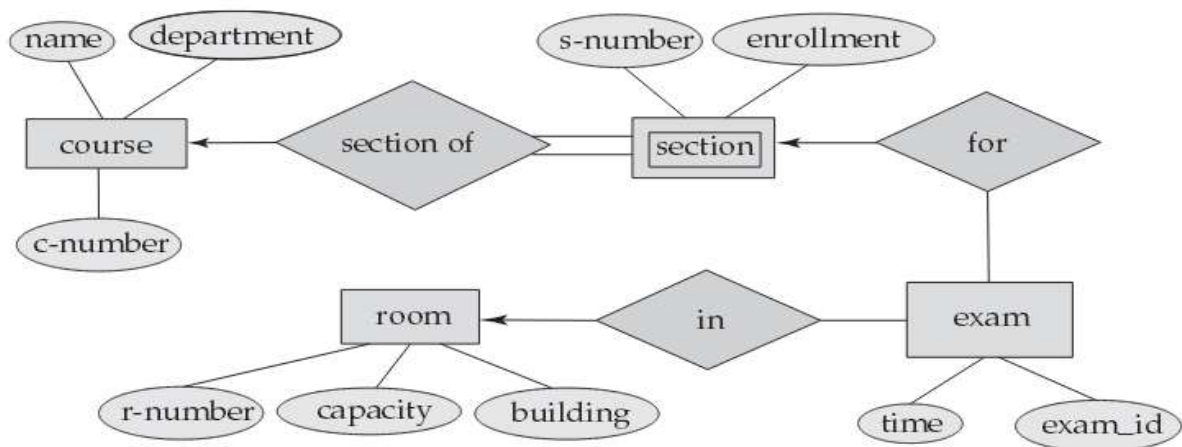


Figure 6.7 E-R diagram for exam scheduling.

8-қадам

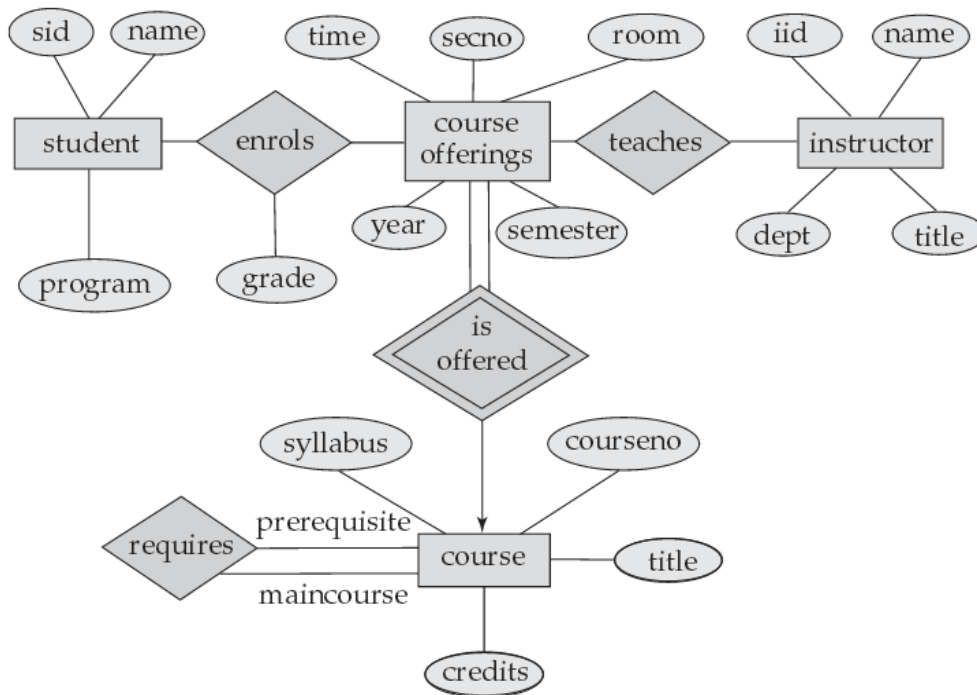


Figure 6.8 E-R diagram for University(a).

9-қадам



Figure 6.9 E-R diagram for University(b).

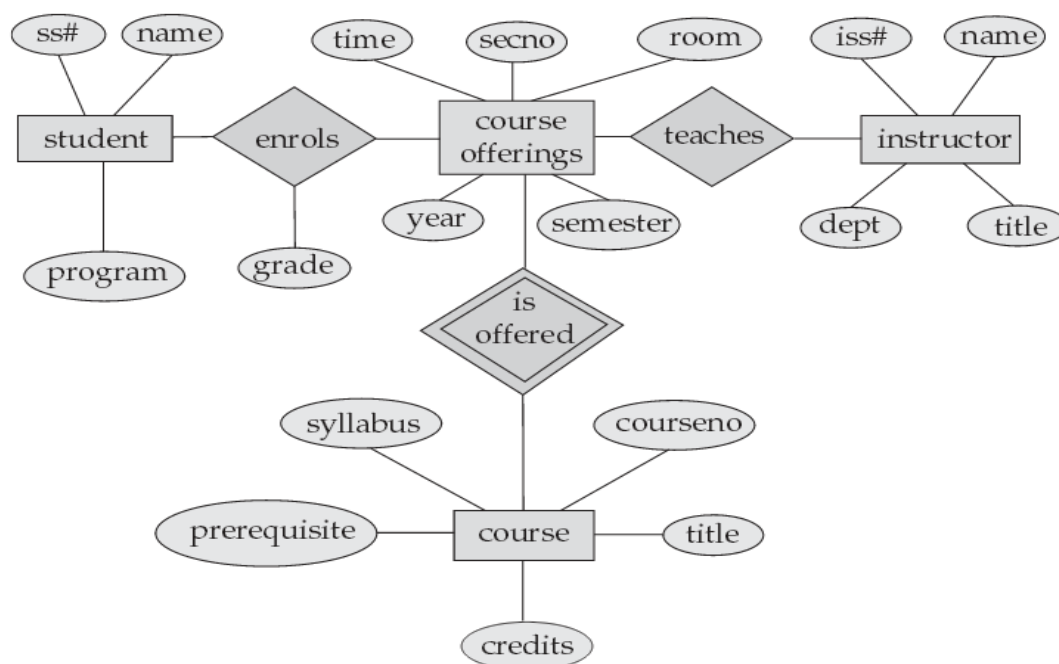


Figure 6.10 E-R diagram for University(c).

Назорат саволлари

1. Диаграммалар асосида қандай соҳа мавжуд?
2. Диаграммаларни қўришда қайси функциялардан фойдаландингиз?
3. Data-definition language функциясини тушунтиринг?
4. View definition функциясини тушунтиринг?
5. SQL Data Definition функциясини тушунтиринг?
6. SQL, Data inconsistency функциясини тушунтиринг?
7. Consistency constraints функциясини тушунтиринг?
8. Data abstraction on Create функциясини тушунтиринг?

Адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. см. 2011. ISBN 978-0-07-352332-3
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)"(2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000)

3-амалий машғулот (2 соат). SQL тили асосида сўровларни амалга ошириш

Ишдан мақсади: SQL оператори ёрдамида танлаш амали орқали сўровларни амалга ошириш, танлаш орқали излаш алгоритмларига асоланган сўровларни яратиш усуллари, саралаш алгоритмлари ва уларнинг қўлланишлари, бирлаштириш алгоритмлари асосида сўровларни қайта ишлаш.

Калит сўзлар: *Сўров, реляцион алгебра, танлаш, оптималлаштириш, баҳолаш, саралаш, излаш, бирламчи ва иккиламчи калит*

Масаланинг қўйилиши: Университет маълумотлар базасида сўровларни қайта ишлаш амалларини бажариш

Ишни бажариш учун намуна

1-қадам. Схемани яратиб оламиз

```
CREATE { PROC | PROCEDURE } [schema_name.] procedure_name [ ; number
]
  [ { @parameter [ type_schema_name. ] data_type }
    [ VARYING ] [ = default ] [ OUT | OUTPUT ] [READONLY]
  ] [ ,...n ]
[ WITH <procedure_option> [ ,...n ] ]
[ FOR REPLICATION ]
AS { [ BEGIN ] sql_statement [;] [ ...n ] [ END ] }
[;]
```

```
<procedure_option> ::=
  [ ENCRYPTION ]
  [ RECOMPILE ]
  [ EXECUTE AS Clause ]
```

2- қадам. Проседура схемаси

--CLR Stored Procedure Syntax

```
CREATE { PROC | PROCEDURE } [schema_name.] procedure_name [ ; number
]
  [ { @parameter [ type_schema_name. ] data_type }
    [ = default ] [ OUT | OUTPUT ] [READONLY]
  ] [ ,...n ]
[ WITH EXECUTE AS Clause ]
AS { EXTERNAL NAME assembly_name.class_name.method_name }
```

[;]

3-қадам. Маълумотларни қайта ишлаш учун жадваллар яратамиз.

```
CREATE PROCEDURE TransProc @PriKey INT, @CharCol CHAR(3)
AS
BEGIN
    INSERT INTO TestTrans VALUES (@PriKey, @CharCol);
    BEGIN TRANSACTION InProc
        PRINT 'In proce begin: ' + CAST(@@TRANCOUNT as char(1));
        INSERT INTO TestTrans VALUES (@PriKey + 1, @CharCol)
    if @@TRANCOUNT > 0
    BEGIN    PRINT 'In proc commit: ' + CAST(@@TRANCOUNT as char(1));
    END;
END;
GO
/* Start a transaction and execute TransProc. */
BEGIN TRANSACTION OutOfProc;
    PRINT 'Out of proc begin: ' + CAST(@@TRANCOUNT as char(1));
    EXEC TransProc 1, 'aaa';
IF @@TRANCOUNT > 0 COMMIT TRANSACTION OutOfProc;
PRINT 'Out of proce commit: ' + CAST(@@TRANCOUNT as char(1));
```

4-қадам

```
CREATE TABLE T1 (c1 INT, c2 CHAR(2))@
CREATE TABLE T2 (c1 INT, c2 CHAR(2))@

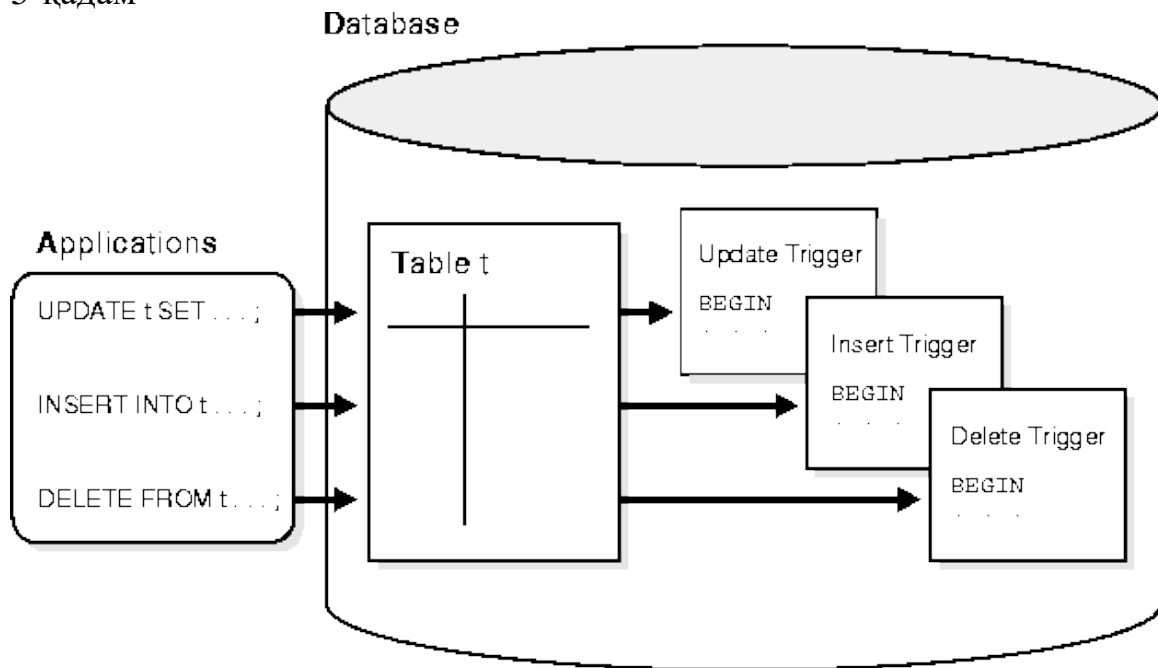
CREATE PROCEDURE proc(IN val INT, IN name CHAR(2))
LANGUAGE SQL
DYNAMIC RESULTSETS 0
MODIFIES SQL DATA
BEGIN
    DECLARE rc INT DEFAULT 0;
    INSERT INTO TABLE T2 VALUES (val, name);
    GET DIAGNOSTICS rc = ROW_COUNT;
    IF ( rc > 0 ) THEN
        RETURN 0;
    ELSE
        RETURN -200;
    END IF;
END@

CREATE TRIGGER trig1 AFTER UPDATE ON t1
REFERENCING NEW AS n
FOR EACH ROW
```

```

WHEN (n.c1 > 100);
BEGIN ATOMIC
  DECLARE rs INTEGER DEFAULT 0;
  CALL proc(n.c1, n.c2);
  GET DIAGNOSTICS rs = RETURN_STATUS;
  VALUES(CASE WHEN rc < 0 THEN RAISE_ERROR('70001', 'PROC CALL
failed'));
END@
5-қадам

```




```
SQL Statement
UPDATE t1 SET ...;
```

Fires the UPDATE_T1 Trigger

```
UPDATE_T1 Trigger
BEFORE UPDATE ON t1
FOR EACH ROW
BEGIN
    .
    .
    INSERT INTO t2 VALUES (...);
    .
    .
END;
```

Fires the INSERT_T2 Trigger

```
UPDATE_T2 Trigger
BEFORE UPDATE ON t2
FOR EACH ROW
BEGIN
    .
    .
    INSERT INTO ... VALUES (...);
    .
    .
END;
```

etc.

REORDER Trigger

<pre>AFTER UPDATE OF parts_on_hand ON inventory</pre>	Triggering Statement
<pre>WHEN (new.parts_on_hand < new.reorder_point)</pre>	Trigger Restriction
<pre> FOR EACH ROW DECLARE /* a dummy variable for counting */ NUMBER X; BEGIN SELECT COUNT(*) INTO X /* query to find out if part has already been */ FROM pending_orders /* reordered-if yes, x=1, if no, x=0 */ WHERE part_n0=:new.part_no; IF x = 0 THEN /* part has not been reordered yet, so reorder */ INSET INTO pending_orders VALUES (new.part_n0, new.reorder quantity, sysdate); END IF; /* part has already been reordered */ END;</pre>	

```

DROP TABLE stat_tab;
CREATE TABLE stat_tab(utype CHAR(8),
    rowcnt INTEGER, uhour INTEGER);

CREATE OR REPLACE PACKAGE stat IS
    rowcnt INTEGER;
END;
/

CREATE TRIGGER bt BEFORE UPDATE OR DELETE OR INSERT ON sal
BEGIN
    stat.rowcnt := 0;
END;
/

CREATE TRIGGER rt BEFORE UPDATE OR DELETE OR INSERT ON sal
FOR EACH ROW BEGIN
    stat.rowcnt := stat.rowcnt + 1;
END;
/

CREATE TRIGGER at AFTER UPDATE OR DELETE OR INSERT ON sal
DECLARE
    typ CHAR(8);
    hour NUMBER;
BEGIN
    IF updating
    THEN typ := 'update'; END IF;
    IF deleting THEN typ := 'delete'; END IF;
    IF inserting THEN typ := 'insert'; END IF;

    hour := TRUNC((SYSDATE - TRUNC(SYSDATE)) * 24);
    UPDATE stat_tab
        SET rowcnt = rowcnt + stat.rowcnt
        WHERE utype = typ
            AND uhour = hour;
    IF SQL%ROWCOUNT = 0 THEN
        INSERT INTO stat_tab VALUES (typ, stat.rowcnt, hour);
    END IF;

EXCEPTION
    WHEN dup_val_on_index THEN
        UPDATE stat_tab
            SET rowcnt = rowcnt + stat.rowcnt
            WHERE utype = typ
                AND uhour = hour;
END;
/

```

Адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. см. 2011. ISBN 978-0-07-352332-3
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)"(2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000)

4-амалий машғулот (2 соат). SQL тили асосида транзакцияни яратиш

Ишдан мақсади: Маълумотлар базаси учун оддий транзакция яратиш ва тестдан ўтказиш. Транзакциянинг моделлари, транзакция орқали сақлаш моделлари, қўшилган транзакциялар ва уларни ишлатиш принциплари, транзакция билан ишловчи SQL операторлари билан ишлаш.

Калит сўзлар: *транзакция, SQL, амаллар кетма – келиги, транзакция хусусиятлари, сақлаш структураси*

Масаланинг қўйилиши: Университет маълумотлар базасида транзакцияларни амалга ошириш ва қайта ишлаш

Ишни бажариш учун намуна

```
BEGIN TRANSACTION
INSERT СОТРУДНИКИ (TAB) VALUES (5)
INSERT ОТРАБОТАНО (TAB) VALUES (5)
IF @@ERROR = 0
COMMIT
ELSE
ROLLBACK
```

```
CREATE PROCEDURE TRANZ
AS BEGIN
BEGIN TRANSACTION
INSERT СОТРУДНИКИ (TAB) VALUES (11)
INSERT ЧАСЫ (TAB) VALUES (11)
IF @@ERROR = 0
```

```

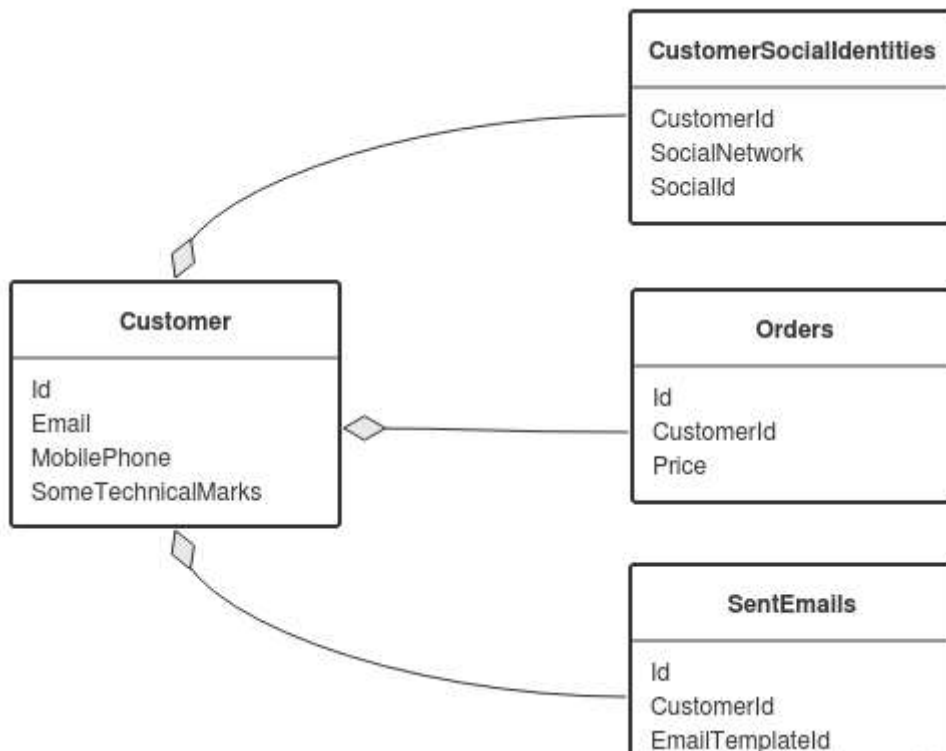
COMMIT
ELSE
ROLLBACK
END

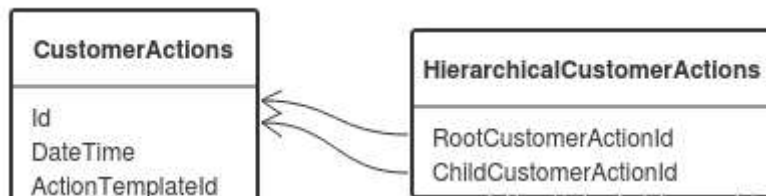
```

```

private const int DefaultRetryCount = 6;
private const int DeadlockErrorNumber = 1205;
private const int LockingErrorNumber = 1222;
private const int UpdateConflictErrorNumber = 3960;
private void RetryOnDeadlock(
    Action<DataContext> action, int retryCount = DefaultRetryCount)
{ if (action == null)
throw new ArgumentNullException("action");
var attemptNumber = 1;
while (true) { var dataContext = CreateDataContext();
try { action(dataContext); break; }
catch (SqlException exception)
{
if (!exception.Errors.Cast<SqlError>().Any(error =>
(error.Number == DeadlockErrorNumber) || (error.Number ==
LockingErrorNumber) || (error.Number == UpdateConflictErrorNumber)))
{ throw; }
else if (attemptNumber == retryCount + 1) { throw; } }
finally { dataContext.Dispose(); } attemptNumber++; }
}

```





```

BEGIN TRAN
UPDATE authors SET au_fname = 'John'
WHERE au_id = '172-32-1176'
UPDATE authors SET au_fname = 'Marg' WHERE au_id = '213-46-8915'
COMMIT TRAN
  
```

```

BEGIN TRAN
UPDATE authors SET au_fname = 'John' WHERE au_id = '172-32-1176'
UPDATE authors SET au_fname = 'JohnY' WHERE city = 'Lawrence' IF
@@ROWCOUNT = 5 COMMIT TRAN
ELSE
ROLLBACK TRAN
  
```

```

Create Proc TranTest1 AS BEGIN TRAN
INSERT INTO [authors]([au_id], [au_lname], [au_fname], [phone], [contract])
VALUES ('172-32-1176', 'Gates', 'Bill', '800-BUY-MSFT', 1)
UPDATE authors SET au_fname = 'Johnzzz' WHERE au_id = '172-32-1176'
COMMIT TRAN GO
  
```

```

Create Proc TranTest2 AS BEGIN TRAN
INSERT INTO [authors]([au_id], [au_lname], [au_fname], [phone], [contract])
VALUES ('172-32-1176', 'Gates', 'Bill', '800-BUY-MSFT', 1)
IF @@ERROR <> 0 BEGIN ROLLBACK TRAN return 10 END
UPDATE authors SET au_fname = 'Johnzzz' WHERE au_id = '172-32-1176' IF
@@ERROR <> 0 BEGIN ROLLBACK TRAN return 11 END
COMMIT TRAN GO
  
```

```
USE pubs
```

```
DECLARE @intErrorCode INT
```

```
BEGIN TRAN
```

```

UPDATE Authors
SET Phone = '415 354-9866'
WHERE au_id = '724-80-9391'
  
```

```
SELECT @intErrorCode = @@ERROR
```

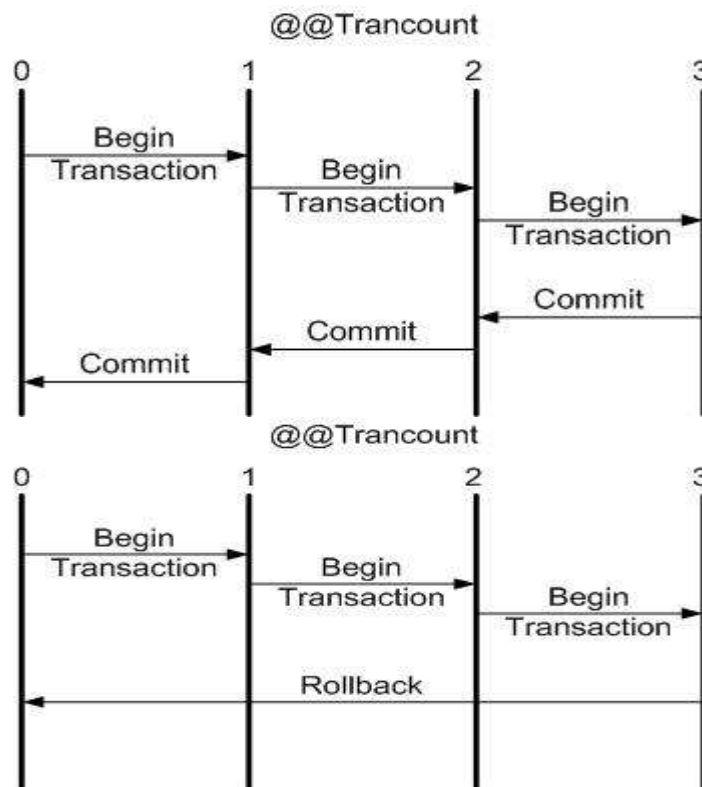
```
IF (@intErrorCode <> 0) GOTO PROBLEM
```

```
UPDATE Publishers  
SET city = 'Calcutta', country = 'India'  
WHERE pub_id = '9999'
```

```
SELECT @intErrorCode = @@ERROR  
IF (@intErrorCode <> 0) GOTO PROBLEM  
COMMIT TRAN
```

PROBLEM:

```
IF (@intErrorCode <> 0) BEGIN  
PRINT 'Unexpected error occurred!'  
ROLLBACK TRAN  
END
```



```
USE pubs  
SELECT 'Before BEGIN TRAN', @@TRANCOUNT -- The value of  
@@TRANCOUNT is 0  
BEGIN TRAN  
SELECT 'After BEGIN TRAN', @@TRANCOUNT -- The value of  
@@TRANCOUNT is 1  
DELETE sales  
BEGIN TRAN nested
```

```

SELECT 'After BEGIN TRAN nested', @@TRANCOUNT
    -- The value of @@TRANCOUNT is 2
DELETE titleauthor
COMMIT TRAN nested
    -- Does nothing except decrement the value of @@TRANCOUNT

SELECT 'After COMMIT TRAN nested', @@TRANCOUNT
    -- The value of @@TRANCOUNT is 1
ROLLBACK TRAN

SELECT 'After ROLLBACK TRAN', @@TRANCOUNT -- The value of
@@TRANCOUNT is 0
-- because ROLLBACK TRAN always rolls back all transactions and sets
-- @@TRANCOUNT to 0.
SELECT TOP 5 au_id FROM titleauthor

```

```

private static void ExecuteSqlTransaction(string connectionString)
{
    using (SqlConnection connection = new SqlConnection(connectionString))
    {
        connection.Open();

        SqlCommand command = connection.CreateCommand();
        SqlTransaction transaction;

        // Start a local transaction.
        transaction = connection.BeginTransaction("SampleTransaction");

        // Must assign both transaction object and connection
        // to Command object for a pending local transaction
        command.Connection = connection;
        command.Transaction = transaction;

        try
        {
            command.CommandText =
                "Insert into Region (RegionID, RegionDescription) VALUES (100,
'Description')";
            command.ExecuteNonQuery();
            command.CommandText =

```



```

        "Insert into Region (RegionID, RegionDescription) VALUES (101,
'Description)";
        command.ExecuteNonQuery();

        // Attempt to commit the transaction.
        transaction.Commit();
        Console.WriteLine("Both records are written to database.");
    }
    catch (Exception ex)
    {
        Console.WriteLine("Commit Exception Type: {0}", ex.GetType());
        Console.WriteLine(" Message: {0}", ex.Message);

        // Attempt to roll back the transaction.
        try
        {
            transaction.Rollback();
        }
        catch (Exception ex2)
        {
            // This catch block will handle any errors that may have occurred
            // on the server that would cause the rollback to fail, such as
            // a closed connection.
            Console.WriteLine("Rollback Exception Type: {0}", ex2.GetType());
            Console.WriteLine(" Message: {0}", ex2.Message);
        }
    }
}
}
}

```

```

using (IDbTransaction tran = conn.BeginTransaction()) {
    try {
        // your code
        tran.Commit();
    } catch {
        tran.Rollback();
        throw;
    }
}

```

```

using(TransactionScope tran = new TransactionScope()) {
    CallAMethodThatDoesSomeWork();
    CallAMethodThatDoesSomeMoreWork();
}

```

```
tran.Complete();  
}
```

```
protected void Button1_Click(object sender, EventArgs e)  
{  
  
    using (SqlConnection connection1 = new SqlConnection("Data  
Source=.\SQLEXPRESS;AttachDbFilename=|DataDirectory|Database.mdf;Integ  
rated Security=True;User Instance=True"))  
    {  
        connection1.Open();  
  
        // Start a local transaction.  
        SqlTransaction sqlTran = connection1.BeginTransaction();  
  
        // Enlist a command in the current transaction.  
        SqlCommand command = connection1.CreateCommand();  
        command.Transaction = sqlTran;  
  
    }  
}
```

Адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. cm. 2011. ISBN 978-0-07-352332-3
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation (2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)" (2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000)

5-амалий машғулот (2 соат). SQL тили асосида маълумотларни йиғиш, жамлаш ва излаш усуллари.

Ишдан мақсади: SQL ва ёрдамчи пакетларнинг вазифалари. Улар ёрдамида маълумотларни сақлаш усуллари, маълумотларни излаш учун индеклаш, параметрли ва зонали индеклаш, қийматни машина ўқуви ёрдамида аниқлаш, **g** нинг энг самарали қиймати, Термин частотаси ва вазни, Тескари маълумотли частота каби излаш усуллари тизимли МВсида қўллаш.

Калит сўзлар: қарор қабул қилиш, *SQL* пакетлари, маълумотларни сақлаш, қоидалар омбори, ечимлар шажараси, энг яхши бўлиниш, маълумотларни излаш, индеклаш, параметрли ва зонали индеклаш, долзарбликнинг вазли зонналари, машина ўқуви, термин частотаси ва вазни, тескари маълумотли частота.

Масаланинг қўйилиши: Университет маълумотлар базасида маълумотларни жамлаш ва излаш усуллари

Ишни бажариш учун намуна

```
public class TransactionDemo
{
    public TransactionDemo()
    {
    }

    [STAThread]
    public static void Main()
    {
        Demo1();
    }

    private static void Demo1()
    {
        SqlConnection db = new SqlConnection("connstringhere");
        SqlTransaction transaction;

        db.Open();
        transaction = db.BeginTransaction();
        try
```

```

{
    new SqlCommand("INSERT INTO TransactionDemo " +
        "(Text) VALUES ('Row1');", db, transaction)
        .ExecuteNonQuery();
    new SqlCommand("INSERT INTO TransactionDemo " +
        "(Text) VALUES ('Row2');", db, transaction)
        .ExecuteNonQuery();
    new SqlCommand("INSERT INTO CrashMeNow VALUES " +
        "('Die', 'Die', 'Die');", db, transaction)
        .ExecuteNonQuery();
    transaction.Commit();
}
catch (SqlException sqlError)
{
    transaction.Rollback();
}
db.Close();
}
}

```

```

// Create the connection and transaction objects.
SqlConnection objConnection = new SqlConnection( strConnectionString );
objConnection.Open();
SqlTransaction objTransaction = objConnection.BeginTransaction();

// Do some stuff here...
SqlCommand objCommand =
    new SqlCommand( "Insert into Table (Name) VALUES ('Rick')",
        objConnection );
objCommand.ExecuteNonQuery();

// Commit the transaction.
objTransaction.Commit();

// Close the connection.
objConnection.Close();
' Create the connection and transaction objects.
Dim objConnection As New SqlConnection( strConnectionString )
objConnection.Open()
Dim objTransaction As SqlTransaction = objConnection.BeginTransaction()

' Do some stuff here...

```

```

Dim objCommand As _
    new SqlCommand( "Insert into Table (Name) VALUES ('Rick')", objConnection )
objCommand.ExecuteNonQuery()

' Commit the transaction.
objTransaction.Commit()

' Close the connection.
objConnection.Close()
// Create the connection and transaction objects.
SqlConnection objConnection = new SqlConnection( strConnectionString );
SqlTransaction objTransaction = null;
try
{
    objConnection.Open();
    objTransaction = objConnection.BeginTransaction();

    // Do some stuff here...
    SqlCommand objCommand =
        new SqlCommand( "Insert into Table (Name) VALUES ('Rick')",
            objConnection );
    objCommand.ExecuteNonQuery();

    // Commit the transaction.
    objTransaction.Commit();
}
catch
{
    if( objTransaction != null )
    {
        objTransaction.Abort();
    }
}
finally
{
    // Close the connection.
    if( objConnection.State == ConnectionState.Open )
        objConnection.Close();
}
}
' Create the connection and transaction objects.
Dim objConnection As New SqlConnection( strConnectionString )
Dim objTransaction As SqlTransaction = Nothing

```

```

Try
objConnection.Open()
objTransaction = objConnection.BeginTransaction()

' Do some stuff here...
Dim objCommand As _
new SqlCommand( "Insert into Table (Name) VALUES ('Rick')",
objConnection )
objCommand.ExecuteNonQuery()

' Commit the transaction.
objTransaction.Commit()
Catch
If objTransaction ,> Nothing Then
objTransaction.Abort()
End If
Finally
' Close the connection.
If objConnection.State = ConnectionState.Open Then
objConnection.Close()
End If
End Try

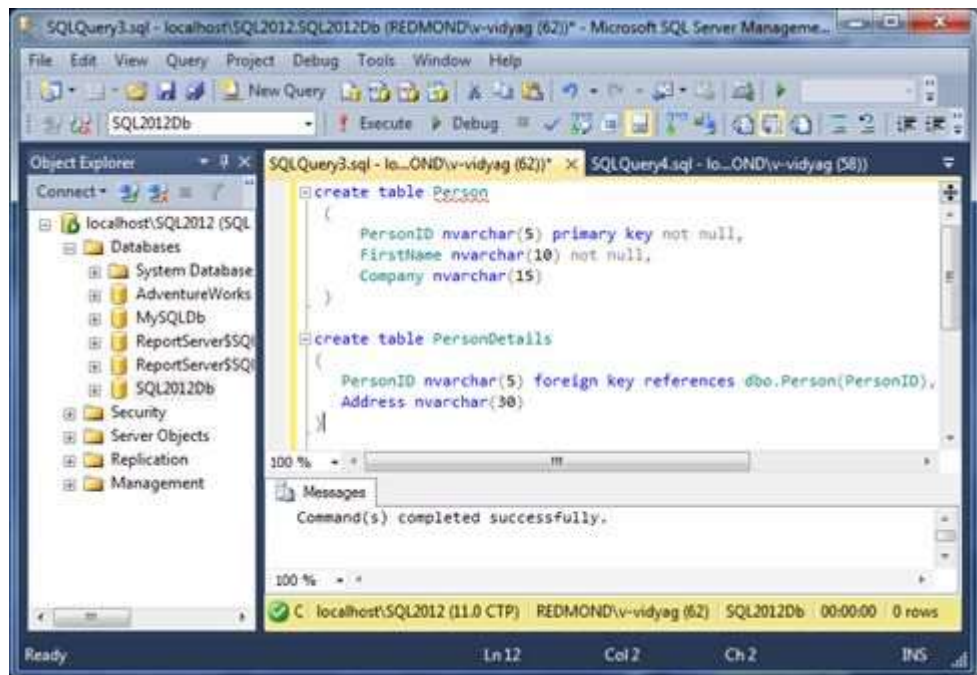
```

```

1.
createtable Person
(
PersonIDnvarchar(5)primary keynot null,
FirstNamenvarchar(10)not null,
Companynvarchar(15)
)

createtable PersonDetails
(
PersonIDnvarchar(5)FOREIGN KEYREFERENCES dbo.Person(PersonID),
Addressnvarchar(30)
)

```

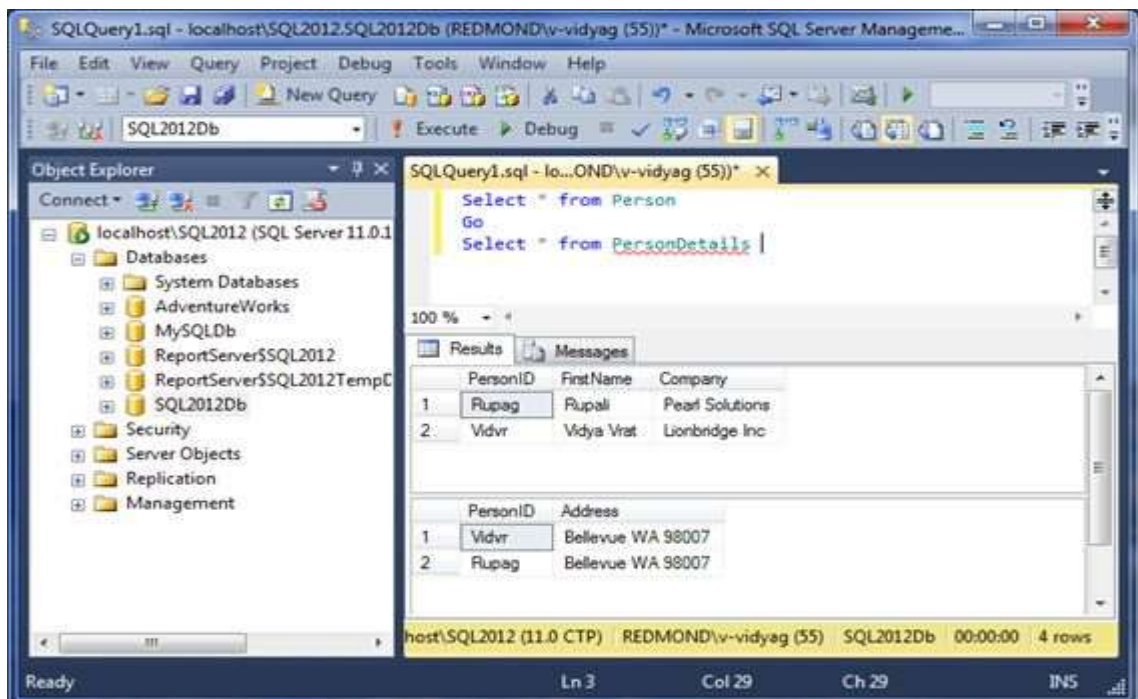


Insertinto Person

values('Vidvr','Vidya Vrat','Lionbridge Inc'),
('Rupag','Rupali','Pearl Solutions')

Insertinto PersonDetails

values('Vidvr','Bellevue WA 98007'),
('Rupag','Bellevue WA 98007')



```

createprocedure sp_Trans_Test

@newpersonidnvarchar(5),
@newfirstnamenvarchar(10)
@newcompanynamenvarchar(15),
@oldpersonidnvarchar(5)
as
declare @inserrint
declare @delerrint
declare @maxerrint

set @maxerr = 0

BEGINTRANSACTION
-- Add a person
insertinto person (personid, firstname, company)
values(@newpersonid, @newfirstname, @newcompanyname)

-- Save error number returned from Insert statement
set @inserr = @@error
if @inserr > @maxerr
set @maxerr = @inserr

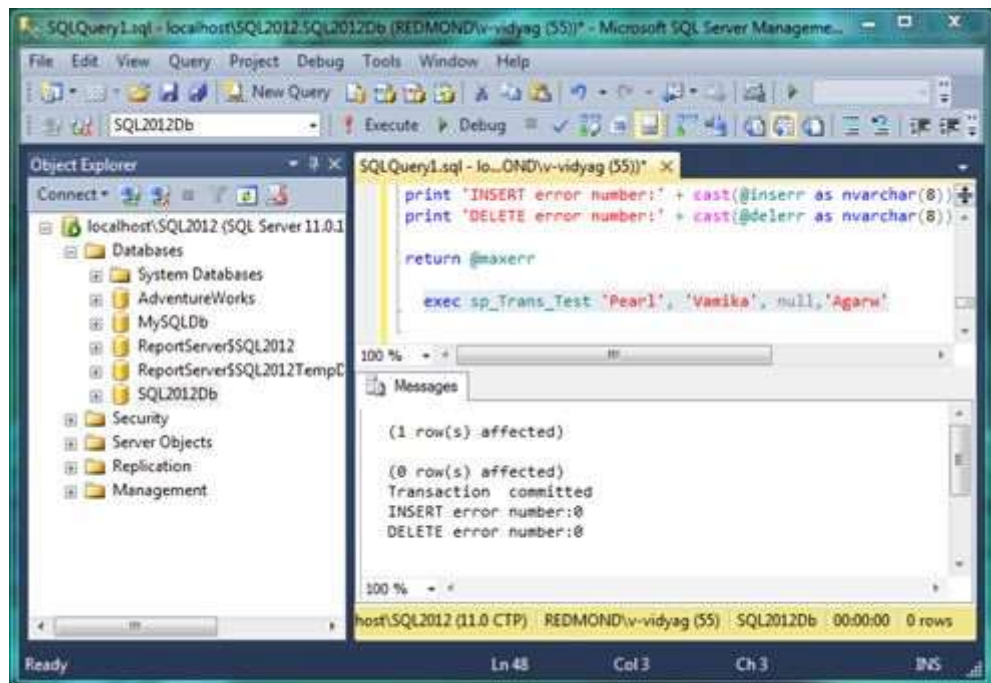
-- Delete a person
deletefrom person
where personid = @oldpersonid

-- Save error number returned from Delete statement
set @delerr = @@error
if @delerr > @maxerr
set @maxerr = @delerr
-- If an error occurred, roll back
if @maxerr <> 0
begin
ROLLBACK
print'Transaction rolled back'
end
else
begin
COMMIT
print'Transaction committed'
end
print'INSERT error number:' + cast(@inserras nvarchar(8))

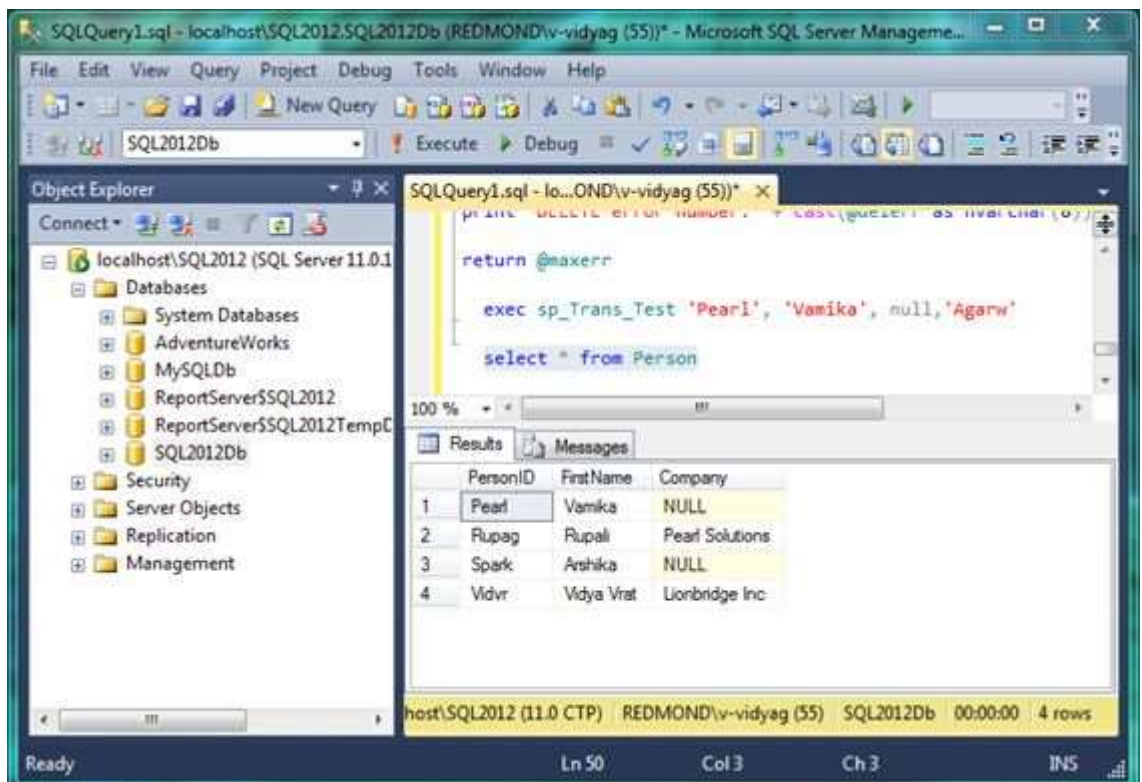
```



```
print 'DELETE error number:' + cast(@delerr as nvarchar(8))
return @maxerr
```



Select* from Person



```

EXEC sp_Trans_Test 'Spark', 'Arshika ', null, 'Agarw'
create procedure sp_Trans_Test
@newpersonid nvarchar(5),
@newfirstname nvarchar(10),
@newcompanyname nvarchar(15),
@oldpersonid nvarchar(5)

as

```

You also declare three local variables:

```

declare @inserr int
declare @delerr int
declare @maxerr int

```

```

BEGIN TRANSACTION

```

```

-- Add a person

```

```

insert into person (personid, firstname, company)
values(@newpersonid, @newfirstname, @newcompanyname)

```

```

-- Save error number returned from Insert statement

```

```

set @inserr = @@error
if @inserr > @maxerr
set @maxerr = @inserr

```

```

-- Delete a person

```

```

delete from person
where personid = @oldpersonid

```

```

-- Save error number returned from Delete statement

```

```

set @delerr = @@error
if @delerr > @maxerr
set @maxerr = @delerr

```

```

-- If an error occurred, roll back

```

```

if @maxerr <> 0
begin
ROLLBACK
print 'Transaction rolled back'
end
else

```

```

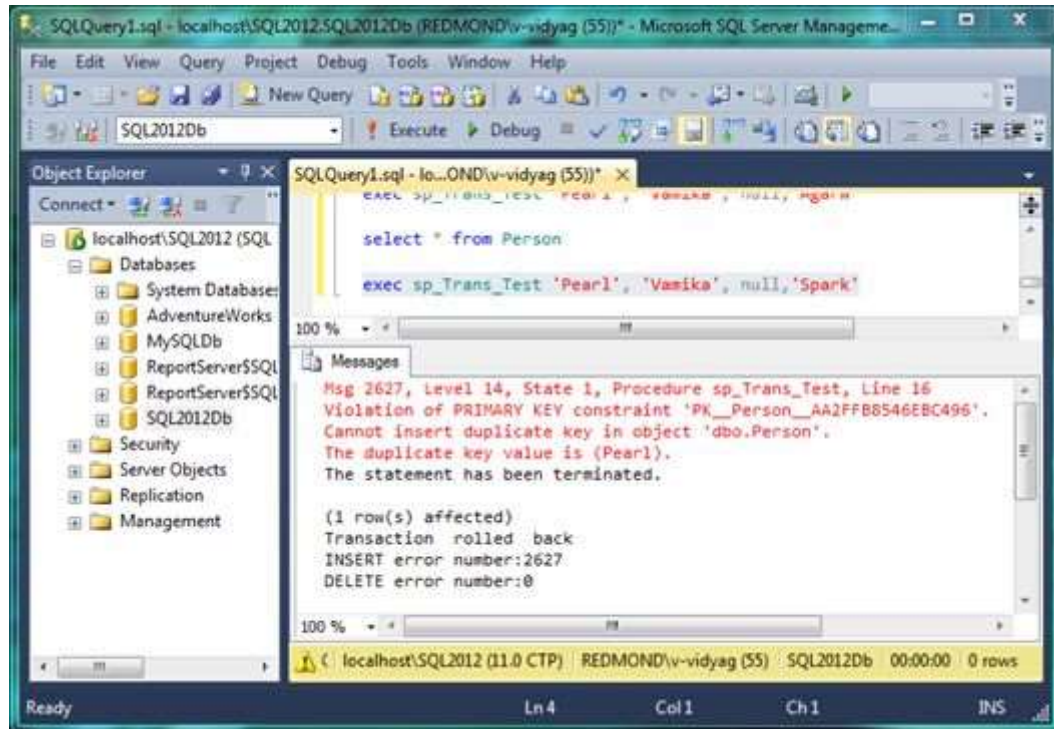
begin
COMMIT
print 'Transaction committed'
end

```

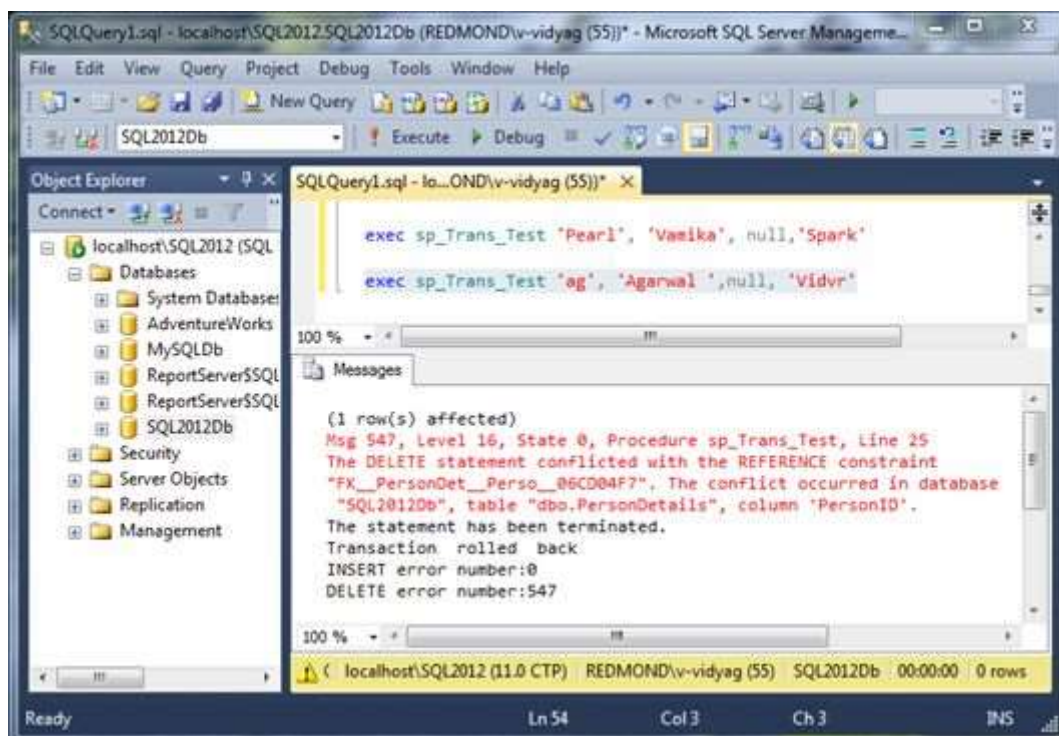
```

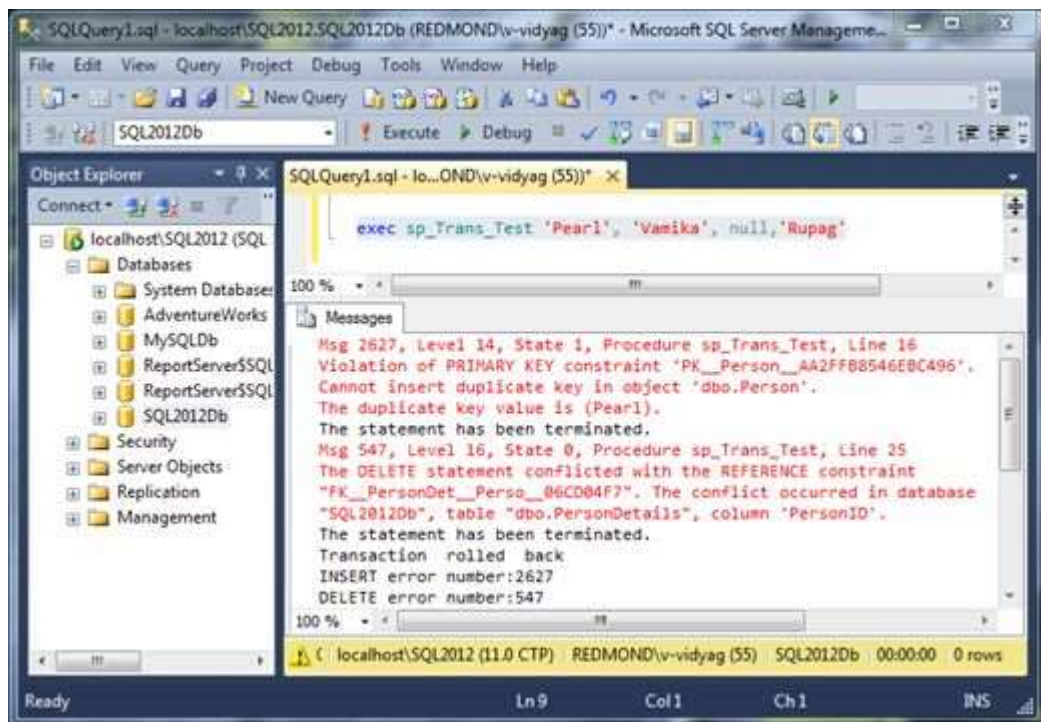
print 'INSERT error number:' + cast(@inserr as nvarchar(8))
print 'DELETE error number:' + cast(@delerr as nvarchar(8))
return @maxerr

```



exec sp_Trans_Test 'ag', 'Agarwal ',null, 'Vidvr'





Адабиётлар ва интернет сайтлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. cm. 2011. ISBN 978-0-07-352332-3
2. N. Walsh et al. "XQuery 1.0 and XPath 2.0 Data Model". <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation (2007).
3. SQL/XML. "ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)" (2004)
4. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann (2000)

У БЎЛИМ

КЕЙСЛАР БАНКИ

V. КЕЙСЛАР БАНКИ

_____ МВБТ НОМИ _____		_____ ЛОЙИҲА НОМИ _____		
_____ ТИНГЛОВЧИНИНГ ИСМИ-ШАРИФИ _____		_____ ИМЗО _____		
ҚИСҚА МАЗМУНИ			БАЛЛ (МАХ)	
ТАШКИЛИЙ ҚИСМ				
Жадваллар учун (ҳар бир жадвал учун 5 баллгача)			25	
Ахборот модели учун (реал боғланганлигини кўрсатиш, тўғри ва тескари)			50	
Ёзувлар сони учун (ҳар бир жадвал учун 10 та ёзув, асосий жадвалда 50 та ёзув)			25	
СИЗ ТЎПЛАГАН БАЛЛ			100	
АМАЛИЙ ҚИСМ				
Т.Р.	ОПЕРАТОР ТУРИ	СЎРОВ СОНИ (МИН)	ҚИСҚА МАЗМУНИ	БАЛЛ (МАХ)
АСОСИЙ ОПЕРАТОРЛАР				
1	INSERT INTO	3	турли типдаги ва бир нечта ёзувларни қўшиш	3
2	SELECT	5	чекланган сондаги ёзувлари чиқариш	3
3	DELETE	2	шарт орқали ўчириш, ҳаммасини ўчириш	2
4	UPDATE	3	турли типдаги ва бир нечта ёзувларни ҳамда ҳаммасини	2
АМАЛЛАР				
5	МАТЕМАТИК АМАЛЛАР	2	камида иккитаси бирга бўлсин	2
6	МУНОСАБАТ АМАЛЛАР	3	камида иккитаси бирга бўлсин	3
7	МАНТИҚИЙ АМАЛЛАР	2	камида иккитаси бирга бўлсин	2
8	АППАЛАШ	2		3
ОПЕРАТОРЛАР				
9	DISTINCT	2		1
10	WHERE	2		1
11	BETWEEN	2		2
12	IN	2		1
13	LIKE	4	фақат бошидан, охиридан, ихтиёрий жойдан	4
14	ISNULL	2	not билан ҳам	1
15	NOT	3	мураккаб амаллар билан	2
16	ORDER BY	2	ҳамма вариантлар учун	1
17	HAVING	2		2
СТАТИК ФУНКЦИЯЛАР				
18	MAX	1		1
19	MIN	1		1
20	COUNT	1		1
21	SUM	1		1
22	COUNT(*)	1		1
23	AVG	1		1
24	АППАЛАШ	3		4
ЁРДАМЧИ СЎРОВЛАР (SUBQUERY - ПОДЧИНЕННЫЕ ЗАПРОСЫ) БИЛАН ИШЛАШ				
25	ЁРДАМЧИ СЎРОВЛАР	3	камида 3 та статик функциялар, амаллар, операторлардан фойдаланг	2
26	IN	2		2

27	EXISTS	2		2
28	ANY	2		2
29	ALL	2		2
МАХСУС ОПЕРАТОРЛАР				
30	CAST	1		1
31	CONVERT	2		2
32	CASE	3		4
33	COALESCE	1		2
34	NULLIF	1		1
БОҒЛАНИШЛАР				
35	ОДДИЙ	2	барча вариантлар	2
36	INNER JOIN	2		2
37	LEFT OUTER JOIN	2		2
38	RIGHT OUTER JOIN	2		2
39	FULL OUTER JOIN	2	барча вариантлар	2
40	CROSS JOIN	2		2
41	UNION	2		3
ЎЗУВ ФУНКЦИЯЛАРИ				
42	FIRST	1		1
43	UCASE	1		1
44	LCASE	1		
45	UPPER	1		
46	LOWER	1		
47	LAST	1		1
48	MID	1		1
49	SUBSTRING	1		1
50	LEN	1		1
51	NOW	1		1
52	САРЛАВҲАНИ НОМЛАШ	1		2
53	FORMAT	1		1
MS ACCESS ДАН БОШҚА МВБТ ТИЗИМЛАРИ УЧУН				
54	COMMIT	2	камида иккита хар хил сўров	2
55	ROLLBACK	2	камида иккита хар хил сўров	2
56	COMMIT & ROLLBACK	2	камида иккита хар хил сўров	2
57	PROCEDURE	2	камида битта сўровли	2
58	QUERY WITH PROCEDURE	2	камида битта сўровли	2
ЖАМИ		107	СИЗ ТЎПЛАГАН БАЛЛ	100
ИЖОДИЙ ЁНДАШИШ				
Амалий машғулотлар жараёнида топшириқларни бажариб кўрсатиши				100
ЖАМИ БАЛЛ (сўз билан) МВ нинг ФАЙЛНОМИ				

Кейсни бажариш босқчилари ва топшириқлар:

- Кейсдаги жами 110 та муаммога мос ҳолда МББ шакллантириш ва ишлаб чиқишда компиляция жараёнини мониторинг қилиш ҳар бир вариант билан бажариш йўллари аниқланг (индивиднал).
- Уларни амалиётда бажариш ва бошқариш имкониятларини кўрсатиб беринг (индивидуал ҳолда).

Амалий топшириқлар

1. Битта ёзув учтадан кўп блоклар билан боғланган бўлсин. Саралаш алгоритмлари билан ишлашни амалга оширинг. Унда куйтдагича маълумотлар берилган бўлсин (kangaroo, 17), (wallaby, 21), (emu, 1), (wombat, 13), (platypus, 3), (lion, 8), (warthog, 4), (zebra, 11), (meerkat, 6), (hyena, 9), (hornbill, 2), (baboon, 12).
2. Берилган SQL сўровни реляцион алгебрага ўтказишни бажаринг.

```
select T.branch name
```

```
from branch T, branch S
```

```
where T.assets > S.assets and S.branch city = "Brooklyn"
```

3. Ички такрорланишлар орқали куйидагиларни ёзинг.

```
branch(branch name, branch city, assets)
```

```
customer (customer name, customer street, customer city)
```

```
loan (loan number, branch name, amount)
```

```
borrower (customer name, loan number)
```

```
account (account number, branch name, balance)
```

```
depositor (customer name, account number)
```

1. Берилган МБси 10та кам бўлмаган амални бажарадиган транзакция яратинг.
2. Берилган МБсида кетма – кет амалга ошириладиган транзакция яратинг.
3. Берилган МБсида навбатма-навбат амалга ошириладиган транзакция яратинг.
4. Берилган МБсида топологик саралашга асоланган транзакцияни амалга оширишга транзакция яратинг.
5. Транзакция маълумотларини log файларга ёзилганлиги ҳақидаги маълумотларни изоҳланг.

VI БЎЛИМ

ГЛОССАРИЙ

VII. ГЛОССАРИЙ

Термин	Ўзбек тилидаги шарҳи	Инглиз тилидаги шарҳи
Маълумотлар базасини бошқариш тизими	Ўзаро боғланган маълумотлар тўплами ва бу маълумотларга боғланишга ёрдам берувчи бириктирилган дастурлар	Data mining is an interdisciplinary subfield of computer science . It is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence , machine learning , statistics , and database systems .
Маълумотлар базасини бошқариш тизимининг асосий мақсади	Маълумотлар базаси маълумотларини сақлаш ва маълумотлардан қулай ва самарали тарзда фойдаланишни таъминлаш. Маълумотлар базаси тизими катта ҳажмдаги маълумотларни бошқариш учун тузилган.	A database is an organized collection of data. It is the collection of schemas, tables, queries, reports, views and other objects. The data are typically organized to model aspects of reality in a way that supports processes requiring information, such as modelling the availability of rooms in hotels in a way that supports finding a hotel with vacancies
SQL(Structured Query Language)	Структуралашган сўровлартили. Аниқ қилиб айтганда SQL – маълумотларни қисм тили, чунки Маълумотлар базасини бошқариш тизими бошқа тил	Structured Query Language is a special-purpose programming language designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data

	воситаларига ҳам эга	stream management system (RDSMS).
DML (Data Manipulation Language)	Маълумотлар билан манипуляциялаш операторлари, маълумотлар базаси биланишлашни таъминлашга мўлжалланган тил ҳисобланади	A data manipulation language (DML) is a family of syntax elements similar to a computer programming language used for selecting, inserting, deleting and updating data in a database. Performing read-only queries of data is sometimes also considered a component of DML.
DDL (Data Definition Language)	Маълумотларни аниқлаш операторлари. DDL операторлар гуруҳи маълумот база объектларини яратиш, такомиллаштириш ва олиб ташлаш учун мўлжалланган	Many data description languages use a declarative syntax to define fields and data types. Structured query language (e.g., SQL), however, uses a collection of imperative verbs whose effect is to modify the schema of the database by adding, changing, or deleting definitions of tables or other objects. These statements can be freely mixed with other SQL statements, making the DDL not a separate language.
Транзакция	Маълумотларни қайта ишлаш операторлар кетма-кетлиги бўлиб, маълумот базаси билан ишлайдиган бўлинмас бирлик ҳисобланади. Транзакция маълумот базасини бир ҳолатдан бошқа ҳолатга ўтказди	In computer science, transaction processing is information processing that is divided into individual, indivisible operations called transactions. Each transaction must succeed or fail as a complete unit; it can never be only partially complete.

<p>Маълумотлар базаси тизимининг архитектураси</p>	<p>Ишлаётган компьютер тизими жуда ката таъсир қилади. Маълумотлар базаси тизими марказлашган ёки клиент-сервер технологияси асосида ишлай олади.</p>	<p>Data BASE architecture is composed of models, policies, rules or standards that govern which data is collected, and how it is stored, arranged, integrated, and put to use in data systems and in organizations.[1] Data is usually one of several architecture domains that form the pillars of an enterprise architecture or solution architecture.</p>
<p>UML</p>	<p>бу машҳур моделлаш тилидир. УМЛ синф диаграммаси синфларни моделлашда кенг қўлланилиб маълумот моделларида умумий мақсад учун фойдаланилади.</p>	<p>The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering, that is intended to provide a standard way to visualize the design of a system.</p>
<p>E- R model</p>	<p>бу реал ҳаётда мавжуд ва бошқаларидан хусусиятига кўра фарқланувчи объектдир. Биз моҳиятни унга таъриф берувчи атрибутлар ёрдамида аниқлаб киритамиз</p>	<p>In software engineering, an entity–relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them.</p>
<p>Маълумотлар-</p>	<p>SQL DDL қалит</p>	<p>A data definition language or</p>

таърифлаш тили (Data-definition language – DDL).	майдонини ўрнатовчи буйруқларни ўз ичига олади. У калит майдонини мантиқий тузилишини бузувчи янги ланишларга йўл қўймайди	data description language (DDL) is a syntax similar to a computer programming language for defining data structures, especially database schemas.
Маълумотларни қайта ишлаш тили (Data-manipulation language – DML).	SQL DML маълумотлар омборидан маълумотларни чақиришни, маълумотлар омборига маълумот киритиш, уларни ўчириш ва маълумотларни ўзгартириш каби амалларни бажаради	A data manipulation language (DML) is a family of syntax elements similar to a computer programming language used for selecting, inserting, deleting and updating data in a database. Performing read-only queries of data is sometimes also considered a component of DML.
Кўринишни аниқлаш (view definition)	SQL DDL кўриниш (view) ларни аниқловчи буйруқларни ўз ичига олади.	In database theory, a view is the result set of a stored query on the data, which the database users can query just as they would in a persistent database collection object.
Транзаксияларни бошқариш (Transaction control)	SQL транзаксияларнинг бошланиши ва тугаши аниқловчи буйруқларни ўз ичига олади.	A Transaction Control Language (TCL) is a computer language and a subset of SQL, used to control transactional processing in a database. A transaction is logical unit of work that comprises one or more SQL statements, usually a group of Data Manipulation Language (DML) statements.

<p>Ўзгармас SQL ва динамик SQL(Embedded SQL and dynamicSQL).</p>	<p>Ўзгармас ва динамик SQL қандай қилиб SQL баёноtlари C, C++ ва Жава каби умумий мақсадли дастурлаштирилмасиз ўрнатилиши мумкинлигини кўрсатади.</p>	<p>Embedded SQL is a method of combining the computing power of a programming language and the database manipulation capabilities of SQL. Embedded SQL statements are SQL statements written inline with the program source code of the host language.</p>
<p>Авторизация(Authorization).</p>	<p>SQL DDL жадваллар ва кўриниш(view)лардан эркин фойдаланиш ҳуқуқини белгиловчи буйруқларини ўзичига олади.</p>	<p>When the SQL standard authorization mode is enabled, object owners can use the GRANT and REVOKE SQL statements to set the user privileges for specific database objects or for specific SQL actions. They can also use roles to administer privileges.</p>
<p>Маълумотлар базаларини бошқариш тизими (МББТ)</p>	<p>database management system (DBMS)рус: системауправлениябаза миданных (СУБД)Умумий ёки махсус мақсаддаги дастурий ва лингвистик воситалар мажмуи.</p>	<p>A database is an organized collection of data. It is the collection of schemas, tables, queries, reports, views and other objects. The data are typically organized to model aspects of reality in a way that supports processes requiring information, such as modelling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.</p>

VII БЎЛИМ

АДАБИЁТЛАР
РЎЙХАТИ

VIII. АДАБИЁТЛАР РЎЙХАТИ

АДАБИЁТЛАР РЎЙХАТИ

I. Ўзбекистон Республикаси Президентининг асарлари

1. Мирзиёев Ш.М. Буюк келажакимизни мард ва олижаноб халқимиз билан бирга қурамиз. – Т.: “Ўзбекистон”, 2017. – 488 б.
2. Мирзиёев Ш.М. Миллий тараққиёт йўлимизни қатъият билан давом эттириб, янги босқичга кўтарамиз. 1-жилд. – Т.: “Ўзбекистон”, 2017. – 592 б.
3. Мирзиёев Ш.М. Халқимизнинг розилиги бизнинг фаолиятимизга берилган энг олий баҳодир. 2-жилд. Т.: “Ўзбекистон”, 2018. – 507 б.
4. Мирзиёев Ш.М. Нияти улуғ халқнинг иши ҳам улуғ, ҳаёти ёруғ ва келажак фааровон бўлади. 3-жилд.– Т.: “Ўзбекистон”, 2019. – 400 б.
5. Мирзиёев Ш.М. Миллий тикланишдан – миллий юксалиш сари. 4-жилд.– Т.: “Ўзбекистон”, 2020. – 400 б.

II. Норматив-ҳуқуқий ҳужжатлар

6. Ўзбекистон Республикасининг Конституцияси. – Т.: Ўзбекистон, 2018.
7. Ўзбекистон Республикасининг 2020 йил 23 сентябрда қабул қилинган “Таълим тўғрисида”ги ЎРҚ-637-сонли Қонуни.
8. Ўзбекистон Республикаси Президентининг 2015 йил 12 июнь “Олий таълим муасасаларининг раҳбар ва педагог кадрларини қайта тайёрлаш ва малакасини ошириш тизимини янада такомиллаштириш чора-тадбирлари тўғрисида”ги ПФ-4732-сонли Фармони.
9. Ўзбекистон Республикаси Президентининг 2017 йил 7 февраль “Ўзбекистон Республикасини янада ривожлантириш бўйича Ҳаракатлар стратегияси тўғрисида”ги 4947-сонли Фармони.
10. Ўзбекистон Республикаси Президентининг 2017 йил 20 апрель “Олий таълим тизимини янада ривожлантириш чора-тадбирлари тўғрисида”ги ПҚ-2909-сонли Қарори.
11. Ўзбекистон Республикаси Президентининг 2018 йил 21 сентябрь “2019-2021 йилларда Ўзбекистон Республикасини инновацион ривожлантириш стратегиясини тасдиқлаш тўғрисида”ги ПФ-5544-сонли Фармони.
12. Ўзбекистон Республикаси Президентининг 2018 йил 19 февраль “Ахборот технологиялари ва коммуникациялари соҳасини янада такомиллаштириш чора-тадбирлари тўғрисида”ги ПФ-5349-сонли Фармони.
13. Ўзбекистон Республикаси Президентининг 2019 йил 27 май “Ўзбекистон Республикасида коррупцияга қарши курашиш тизимини янада такомиллаштириш чора-тадбирлари тўғрисида”ги ПФ-5729-сон Фармони.
14. Ўзбекистон Республикаси Президентининг 2019 йил 17 июнь

“2019-2023 йилларда Мирзо Улуғбек номидаги Ўзбекистон Миллий университетида талаб юқори бўлган малакали кадрлар тайёрлаш тизимини тубдан такомиллаштириш ва илмий салоҳиятини ривожлантири чора-тадбирлари тўғрисида”ги ПҚ-4358-сонли Қарори.

15. Ўзбекистон Республикаси Президентининг 2019 йил 27 август “Олий таълим муассасалари раҳбар ва педагог кадрларининг узлуксиз малакасини ошириш тизимини жорий этиш тўғрисида”ги ПФ-5789-сонли Фармони.

16. Ўзбекистон Республикаси Президентининг 2019 йил 8 октябрь “Ўзбекистон Республикаси олий таълим тизимини 2030 йилгача ривожлантириш концепциясини тасдиқлаш тўғрисида”ги ПФ-5847-сонли Фармони.

17. Ўзбекистон Республикаси Вазирлар Маҳкамасининг 2019 йил 23 сентябрь “Олий таълим муассасалари раҳбар ва педагог кадрларининг малакасини ошириш тизимини янада такомиллаштириш бўйича кўшимча чора-тадбирлар тўғрисида”ги 797-сонли Қарори.

18. Ўзбекистон Республикаси Президентининг 2019 йил 21 май “«Электрон ҳукумат» тизими доирасида ахборот-коммуникация технологиялари соҳасидаги лойиҳаларни ишлаб чиқиш ва амалга ошириш сифатини яхшилаш чора-тадбирлари тўғрисида”ги ПҚ-4328-сонли Қарори.

19. Ўзбекистон Республикаси Президенти Шавкат Мирзиёевнинг 2020 йил 25 январдаги Олий Мажлисга Мурожаатномаси.

20. Ўзбекистон Республикаси Президентининг 2020 йил 29 октябрь “Илм-фанни 2030 йилгача ривожлантириш концепциясини тасдиқлаш тўғрисида”ги ПФ-6097-сонли Фармони.

21. Ўзбекистон Республикаси Президентининг 2020 йил 5 октябрь “Рақамли Ўзбекистон-2030” Стратегиясини тасдиқлаш ва уни самарали амалга ошириш чора-тадбирлари тўғрисида”ги ПФ-6079-сонли Фармони.

III. Махсус адабиётлар

1. Silberschatz, Abraham. Database system concepts / Abraham Silberschatz. 6th ed. p. cm. 2011. ISBN 978-0-07-352332-3

2. N. Walsh et al. “XQuery 1.0 and XPath 2.0 Data Model”. <http://www.w3.org/TR/xpath-datamodel>. currently a W3C Recommendation(2007).

3. SQL/XML. “ISO/IEC 9075-14:2003, Information Technology: Database languages: SQL. Part 14: XML-Related Specifications (SQL/XML)”

IV. Интернет сайтлар

22. <http://www.mitc.uz>

23. <http://lex.uz>

24. <http://bimm.uz>

25. <http://ziyonet.uz>

26. <http://natlib.uz>
27. <http://www.tuit.uz>
28. <http://www.db-book.com/> **Database System Concepts**
29. https://en.wikipedia.org/wiki/Database_System_Concepts
30. <http://highereducation.com/sites/0073523321/index.html>

РЕЦЕНЗИЯ

на учебно-методический комплекс, составленный профессором Б.Муминовым по модулю «Создание баз данных на языке запросов SQL» для курсов повышения квалификации и переподготовки педагогических кадров высших образовательных учреждений направления «Программный инжиниринг»

Учебно-методический комплекс по модулю «Создание баз данных на языке запросов SQL» составлен для курсов повышения квалификации и переподготовки педагогических кадров высших образовательных учреждений направления «Программный инжиниринг» и содержит в себе программу курсов, рекомендованные педагогические технологии, тексты лекций, материалы для практических занятий, кейсы, глоссарий и список рекомендованной литературы и интернет сайтов.

Программа модуля соответствует содержанию типовой программы и включает в себя введение, цели и задачи модуля, требования к знаниям, умениям, навыкам и компетенциям слушателей, рекомендации к проведению занятий, разбивка часов по темам, краткое содержание теоретических и практических занятий, список рекомендованной литературы и интернет сайтов.

Разработанный учебно-методический комплекс по модулю «Создание баз данных на языке запросов SQL» соответствует содержанию типовой и учебной программы, часы распределены соответственно часам, указанным в учебном плане.

Подводя итог, можно сказать, что учебно-методический комплекс по модулю «Создание баз данных на языке запросов SQL» может быть рекомендован к использованию на курсах повышения квалификации и переподготовки педагогических кадров высших образовательных учреждений направления «Программный инжиниринг» и его можно рекомендовать к публикации.

И.о. заместителя директора по научной работе и инновациям Совместного Белорусско-Узбекского межотраслевого института прикладных технических квалификаций, к.п.н.


Набиуллина
подпись завершено,
кабинет № 101,
Мини-аппарат
16.12.2020г.

**ОЛИЙ ТАЪЛИМ МУАССАСАЛАРИ ПЕДАГОГ КАДРЛАРИНИ
ҚАЙТА ТАЙЁРЛАШ ВА МАЛАКАСИНИ ОШИРИШ КУРСИ УЧУН
ТАЙЁРЛАНГАН “SQL СЎРОВЛАР ТИЛИДА МАЪЛУМОТЛАР
БАЗАЛАРИНИ ЯРАТИШ” МОДУЛИНИНГ ЎҚУВ-УСЛУБИЙ
МАЖМУАСИГА**

ТАҚРИЗ

Ўқув-услубий мажмуа “SQL сўровлар тилида маълумотлар базаларини яратиш” модули бўйича қайта тайёрлаш ва малака ошириш тингловчилари учун яратилган. Қайта тайёрлаш ва малака ошириш йўналишининг ўзига хос хусусиятлари ҳамда долзарб масалаларидан келиб чиққан ҳолда ўқув-услубий мажмуада тингловчиларнинг “SQL сўровлар тилида маълумотлар базаларини яратиш” модули доирасидаги билим, кўникма, малака ҳамда компетенцияларига қўйиладиган талаблар асосида ўқув-услубий мажмусида берилган материаллар ушбу мақсадга йўналтирилиб, маълумотлар базасининг фундаментал назанияси ва SQL сўровлар тилида маълумотлар базаларини яратиш хусусиятларини ўрганиш, уларни таълим жараёнига қўллаш бўйича назарий ва амалий маълумотлар келтирилган.

Ўқув-услубий мажмуа доирасида берилаётган мавзулар таълим соҳаси бўйича педагог кадрларни қайта тайёрлаш ва малакасини ошириш мазмуни, сифати ва уларнинг тайёргарлигига қўйиладиган умумий малака талаблари, ўқув режалари ва дастурлари асосида шакллантирилган бўлиб, бу орқали олий таълим муассасалари педагог кадрларининг соҳага оид замонавий таълим ва инновация технологиялари, илғор хорижий тажрибалардан самарали фойдаланиш, киберхавфсизлик усул ва воситаларини амалиётга кенг татбиқ этиш билан боғлиқ компетенцияларга эга бўлишлари таъминланади.

Умуман олганда, “SQL сўровлар тилида маълумотлар базаларини яратиш” модули бўйича яратилган ўқув-услубий мажмуа барча талабларга жавоб беради ва уни ўқув жараёнида қўллаш ва чоп этиш учун тавсия этиш мумкин.

Муҳаммад Ал-Хоразмий номидаги
ТАТУ “Ахборот технологиялари” кафедраси
мудир, профессор



Х.Зайнидинов

