

№ 5 дәріс. Деректер қорларының жүйелері

Дәрістің мақсаты: деректер қоры жүйесінің негіздерін меңгеру.

Дәрістің мазмұны: Деректер қоры жүйесінің негіздері: ұғымы, сипаттамасы, архитектурасы. Деректер моделі. Қалыпқа келтіру. Деректер тұтастығына шектеу. Сұраныстарды тиімділеу және оларды өңдеу. SQL негіздері. Деректерді параллельді өңдеу және оларды қалпына келтіру. Деректер қорын жобалау және құру. ORM бағдарламалау технологиясы. Үлестірілген, параллельді және гетерогенді деректер базасы.

5.1 Деректер қорының жүйесі туралы негізгі түсініктер

Деректер - жинауды, сақтауды және адам немесе ақпараттық құрылғы арқылы одан әрі өңдеуді автоматтандыруға мүмкіндік беретін белгілі бір түрде ұсынылған ақпарат. Компьютерлік технологиялар үшін деректер - бұл дискретті, бекітілген түрдегі, ЭЕМ сақтауға, өңдеуге, сондай-ақ байланыс арналары арқылы таратуға ыңғайлы ақпарат.

Деректер қоры (ДҚ) – объектілердің жай-күйін және олардың қарастырылып отырған пәндік саладағы қатынастарын көрсететін аталған мәліметтер жиынтығы. ДҚ көптеген байланысты файлдардан тұрады.

Деректер қорын басқару жүйесі (ДҚБЖ) – бұл көптеген пайдаланушылар үшін ДҚ құруға, жүргізуге және бөлісуге арналған тілдік және бағдарламалық құралдардың жиынтығы. Жобаланған мәліметтер қоры келесі қасиеттерге ие болуы керек:

- *тұтастық*, шектеулерді енгізу нәтижесінде қол жеткізілетін тұтастық, атап айтқанда, оларға мәліметтер базасын қалыпқа келтіруге байланысты шектеулер жатады;

- жүйенің істен шығуынан немесе жүйенің бұзылуының жекелеген түрлерінен кейін *ДҚ қалпына келтіру*;
- *ДҚ қауіпсіздігі*, деректерді қасақана және байқаусызда қол жеткізуден, өзгертуден немесе жоюдан, ұрлаудан қорғауды көздейді;
- *тиімділік*: пайдаланушының сұранысына жауап берудің минималды уақыты, жадының минималды қажеттілігі.

ДҚБЖ дамытудың маңызды аспектілерінің бірі – ДҚ-ның логикалық құрылымын бөлу және пайдаланушыларға қажетті мәліметтерді компьютерлік жабдық қажет ететін физикалық көріністен манипуляциялау идеясы болып табылады.

ДҚБЖ қолдайтын мәліметтерді сипаттау деңгейлерінің санына сәйкес бір, екі және үш деңгейлі жүйелерге бөлінеді. Қазіргі уақытта деректер қорын сипаттаудың үш деңгейлі архитектурасы жиі қолданылады.

Үш деңгейлі архитектура мыналарды қамтиды:

- пайдаланушылардың немесе пайдаланушылардың жекелеген топтарының өздері құратын дерекқорға өз көзқарасы болатын *сыртқы деңгейі*;
- ДҚБЖ және операциялық жүйе деректерді қабылдайтын *ішкі деңгейі*;
- сыртқы деңгейді ішкі деңгейге көрсетуге, сондай-ақ олардың бір-бірінен қажетті тәуелсіздігін қамтамасыз етуге арналған деректерді ұсынудың *тұжырымдамалық деңгейі*; бұл пайдаланушылардың жалпыланған көрінісімен байланысты.

Кез-келген деңгейдегі мәліметтер құрылымының сипаттамасы *схема* деп аталады.

Деректер моделі – бұл ақпарат бірліктерінің құрылымын және олардың ақпараттық жүйедегі операцияларын формальды сипаттау.

Деректер моделі – бұл таңдалған пән саласының жұмысының маңызды аспектілерін көрсететін абстракцияның бір түрі, ал екінші кезектегілері еленбейді. Деректер моделі деректерді сипаттауға арналған ұғымдар жиынтығын, олардың арасындағы байланыстарды және деректерге қойылған шектеулерді қамтиды.

Деректер моделінде *үш негізгі құраушы* бар:

- ДҚБЖ үшін рұқсат етілген деректер құрылымдарының түрлерін құру ережелерін анықтайтын *құрылымдық бөлік*;
- құрылымдарда мүмкін болатын операцияларды анықтайтын *басқару бөлігі*;
- осы жүйенің құралдарымен жүзеге асырылатын *деректердің тұтастығын шектеу кластары*.

Әрбір ДҚБЖ белгілі бір деректер моделін қолдайды. Негізінде, ДҚБЖ механизмдері қолдайтын деректер моделі осы жүйенің көмегімен жасалуы мүмкін көптеген нақты деректер қорларды анықтайды, сондай-ақ пәндік салада болып жатқан өзгерістерді көрсету үшін ДҚ күйін өзгерту жолдарын толығымен анықтайды.

Бүгінгі таңда көптеген әртүрлі модельдер сипатталған, және олардың құрылысы әртүрлі мақсаттарды көздейді. Жарияланған көптеген мәліметтер үлгілерінің ішінен үш санатты ажыратуға болады:

- деректердің объектілік үлгілері (ER үлгісі, API);
- жазбаларға негізделген деректер үлгілері (желілік, иерархиялық);
- деректердің физикалық үлгілері (жалпылама үлгі және кадрлық жад үлгісі).

Қалыпқа келтіру - бұл деректер қорындағы деректерді ұйымдастыру процесі, ол кестелер құруды және олардың арасындағы қатынастарды ережелерге сәйкес орнатуды қамтиды, бұл деректерді қорғауды қамтамасыз етеді және деректер қорын ыңғайлы етеді, артық және сәйкес келмейтін тәуелділіктерді жояды. Деректердің артықтығы дискідегі бос орынның өнімсіз жұмсалыуына әкеледі және деректер қорына қызмет көрсетуді қиындатады.

Тұтастықты шектеу – бұл деректік қорларға жарамсыз деректерге қол жеткізуді шектейтін арнайы құралдардың жиынтығы (мысалы, деректерді енгізу кезінде пайдаланушылардың қателіктерін ескерту).

Тұтастық шектеулері үш санат (категорияға) бөлінеді:

а) *бірінші санат* – домендік тұтастықты қамтамасыз ету құралдары;

б) *екінші санат* – мәндік тұтастық;

в) *үшінші санат* – анықтамалық тұтастық, бастапқы және сыртқы кілттер жүйесімен қамтамасыз етіледі.

Тұтастықты қамтамасыз ету құралдарын бөлуге болатын тағы екі үлкен санат - бұл *декларативті және процедуралық* сипаттағы құралдар. *Декларативті сипаттағы құралдар* деректер базасында анықталған кезде объектілердің құрамдас бөліктері ретінде құрылады (мысалы, мәліметтер қорындағы кестені анықтаудағы мән шарты). *Процуралық құралдар* (триггерлер және сақталған процедуралар) жеке бағдарламалық модульдер ретінде жүзеге асырылады. Жалпы алғанда, декларативті шектеулер аз функционалды, бірақ ресурстарды тиімді пайдаланады және керісінше.

Сұраныс – бұл деректер базасынан алынатын мәліметтерді сипаттайтын тілдік өрнек. Сұраныстарды тиімділеу контекстінде сұраныстар мазмұнға негізделген (көп жағдайда жиынтыққа бағытталған) түрде көрсетілетіні түсініледі, бұл тиімділеушіге есептеу процедуралары арасында таңдау жасауға мүмкіндік береді.

Сұраныстарды тиімділеу берілген сұраныстың жауап беру уақытын және берілген жүйелік ортадағы сұраныс түрлерінің қосындысын азайтуға бағытталған. Сұраныстарды өңдеу эвристикалық әдісті және сұранысты орындаудың әртүрлі нұсқаларының құнын салыстырмалы бағалаудан және ресурстарды пайдаланудың минимумын таңдаудан тұратын әдісті қолдануды қамтиды

ДҚ-нің бірнеше түрі бар, олардың ішінде:

1) *үлестірілген деректер қоры* (DDB – distributed database) – бұл компьютерлік желіде таратылған логикалық өзара байланысы бар деректер қорының жиынтығы. Үлестірілген деректер қорын басқару жүйесі үлестірілген дерекқорды оның таралуы пайдаланушылар үшін ашық болатындай басқаруға мүмкіндік беретін бағдарламалық жүйе ретінде анықталады.

2) *Параллель ДҚБЖ* кең ауқымды параллелизмді қолдана отырып, мультипроцессорлық компьютерде жүзеге асырылады. Бұл анықтама көптеген баламалардың болуын болжайды, олардың спектрі тек интерфейсті операциялық жүйеге қайта өңдеумен қолданыстағы ДҚБЖ тікелей тасымалдаудан бастап, параллельді өңдеу алгоритмдерінің күрделі комбинацияларына және жаңа аппараттық-бағдарламалық архитектураларға әкелетін мәліметтер базасының функцияларына дейін ауытқиды.

3) *гетерогенді МҚБЖ* – бұл бөлінген ДҚБЖ-нен тұратын ДҚБЖ-ның әртүрлі типтері.

5.2 SQL негіздері

SQL тілі (Structured Query Language – құрылымдық сұраныс тілі) – реляциялық мәліметтер үлгісіне негізделген деректер қорын басқару жүйелеріндегі (ДҚБЖ) деректерді сипаттауға және өңдеуге арналған стандартты жоғары деңгейлі тіл болып табылады.

SQL тілі – деректер кестелерімен жұмыс істеуге бағытталған және ол күрделі бағдарламаларды жүзеге асыру үшін жеткілікті құралдарға ие емес. Сондықтан, ол C++, Паскаль сияқты жоғары деңгейлі тілдермен бірге қолданылады, немесе арнайы әзірленген ДҚБЖ тілінің құрамына кіреді.

SQL тілінде әзірленген "бағдарлама" – бұл SQL тілінің операторларының қарапайым сызықтық тізбегі. SQL тілінде өзінің "таза" түрінде мәліметтер қорына сұраныстардың орындалу ретін басқаруға арналған операторлары жоқ (цикл, тармақ, ауысу түрі). SQL тілінің операторлары мыналарды қолдану арқылы құрылады:

- резервтелген *кілттік сөздер*;
- кестелер мен кесте бағандарының *идентификаторлары (атаулары)*;
- мәліметтер қорында ақпаратты іздеу критерийлерін қалыптастыру және алынған кестелер ұяшықтарының мәндерін есептеу үшін қолданылатын *логикалық, арифметикалық және жолдық өрнектер*;
- өрнектерде қолданылатын операциялар мен функциялардың *идентификаторлары (атаулары)*.

Барлық кілттік сөздер, функция атаулары және кесте мен баған атаулары 7 биттік ASCII кодтау таңбаларымен (басқаша айтқанда, латын әріптерімен) ұсынылады. SQL тілінде Бас (үлкен) және кіші (кіші) әріптер арасында ешқандай айырмашылық жоқ, мысалы, «SELECT», «Select», «select» жолдары бірдей кілттік сөзді білдіреді. Кесте атаулары мен олардың бағандарын құру үшін әріптерді, сандарды және "_" белгісін (астын сызу) қолдануға болады, бірақ атаудың бірінші таңбасы әріп болуы керек. Кілттік сөздер мен функция атауларын кесте идентификаторы және баған атаулары ретінде пайдалануға тыйым салынады. Кілттік сөздер мен функция атауларының толық тізімін (және ол өте кең) белгілі бір ДҚБЖ құжаттамасынан табуға болады. Оператор кілттік сөзден-етістіктен басталады (мысалы, «CREATE» – құру, «UPDATE» – жаңарту, «SELECT» – таңдау және т.б.) және ";" (нүктелі үтір) белгісімен аяқталады. Оператор еркін форматта жазылады және бірнеше жолды алады.

Деректер базасын жобалаудың бірнеше кезеңдері бар:

- 1) тапсырмаларды нақтылау.
- 2) тапсырмаларды орындау реттілігі.
- 3) деректерді талдау.
- 4) деректер құрылымын анықтау.
- 5) қолданба макетін және пайдаланушы интерфейсін әзірлеу.
- 6) қосымшаны құру.
- 7) тестілеу және жетілдіру.

5.3 ORM бағдарламалау технологиясы

ORM (object–relational mapping – объектілік-реляциялық бейнелеу) – "виртуалды нысан деректер қорын" құру арқылы бағдарламалау тілін біріктіретін бағдарламалау технологиясы.

ORM-дің міндеті - мәліметтер қорын жазуда және кері бағытта объект типтерін таратуды басқару. Негізгі мәселе – ондағы объектілер иерархиялық құрылымға ие, ал мәліметтер базасы реляциялық құрылымға ие.

Жобада ORM пайдалану әзірлеушіні SQL-мен жұмыс істеу және көптеген код жазу қажеттілігінен құтқарады. ORM жасаған барлық код жақсы тексерілген және оны тестілеу туралы ойланудың да қажеті жоқ. ORM-дің кемшілігі – бұл жүйенің мәліметтерді пайдалану сценарийлері кең спектрді өңдейді, соған байланысты өнімділік жоғалады, бұл кез-келген жеке қосымшадан әлдеқайда көп.

5.4 бақылау сұрақтары

1. ДҚ дегеніміз не?
2. Деректер дегеніміз не?
3. ДҚ түрлері қалай жіктеледі?
4. Сұраныс дегеніміз не?
5. Қандай мәліметтер үлгілері бар?
- 5 ORM дегеніміз не?
- 6 SQL тілінің негіздері?
- 7 Сұраныстарды тиімділеу дегеніміз не?
- 8 Қалыпқа келтіру дегеніміз не?
- 9 ДҚ - на қандай талаптар қойылады?