

Тақырып 3. ДЕРЕКТЕР БАЗАСЫН ҚҰРУ ПРИНЦИПТЕРІ.

Реляциялық дерекқорды жобалаудың классикалық технологиясы қатынастардың атрибуттары арасындағы функционалдық тәуелділіктерді талдауға негізделген қалыпқа келтіру теориясымен байланысты. Қалыпқа келтіру процесі деректердің артықтығын жоюға бағытталған. Қалыпқа келтіру сақталған ақпарат көлемін едәуір азайтуға және деректерді сақтауды ұйымдастырудағы ауытқуларды жоюға мүмкіндік береді. Деректерді қалыпқа келтіру дәрежесі әртүрлі болуы мүмкін. Модельді қалыпты форманың қажетті деңгейіне келтіру реляциялық мәліметтер базасын құрудың негізі болып табылады.

Қалыпқа келтіру кестелердің қалыпқа келтірудің үш деңгейінде анықталған бірқатар шарттарға сәйкестігін тексеру арқылы жүзеге асырылады: бірінші, екінші және үшінші қалыпты нысандар (басқа деңгейлер де бар).

Бірінші қалыпты форма ДБ кестесінің әр өрісі бөлінбейтіндігін және қайталанатын топтар болмауын талап етеді.

Өрістің бөлінбеуі оның құрамындағы мәндер кіші мәндерге бөлінбеуі керек дегенді білдіреді. Мысалы, егер "бөлім" өрісінде факультет пен кафедраның атауы болса, бөлінбеу талабы сақталмайды және факультеттің немесе кафедраның атауын бөлек өріске бөлу қажет.

Қайталанатын-мағынасы бірдей мәндер бар өрістер. Мысалы, егер сіз ай сайын төрт өнімді сату статистикасын алғыңыз келсе, әр өнім үшін сату деректерін сақтау үшін өрістер жасай аласыз. Алайда, егер тауарлар 4 емес, 104 болса, не істеу керек және тауарлардың саны алдын-ала белгісіз болса ше? Қайталанатын топтарды кестеде жалғыз "өнім"өрісін сақтау арқылы жою керек. Нәтижесінде біз бір өнімді сату статистикасы туралы ақпаратты қамтитын жазбаны аламыз, бірақ бұл өнім кез-келген болуы мүмкін.

Екінші қалыпты форма кестенің барлық өрістерінің бастапқы кілтке тәуелді болуын талап етеді, яғни бастапқы кілт жазбаны бірегей түрде анықтайды және артық болмайды. Егер кез-келген кестеде негізгі емес өрістердің бастапқы кілттің бөлігіне тәуелділігі болса, оларды жеке кестеге бөліп, жаңа кестенің бастапқы кілтін осы өрістерге тәуелді болатын бастапқы

кілттің бір бөлігіне айналдырып, "бір-көп" байланысын орнату керек.жаңа кестеден ескіге дейін.

Үшінші қалыпты форма кестелерде негізгі емес өрістер арасында өтпелі тәуелділіктер болмауын талап етеді, яғни бастапқы кілтке кірмейтін кез-келген өрістің мәні бастапқы кілтке кірмейтін басқа өрістің мәніне тәуелді болмайды.

Нормализацияның нәтижесі-бұл деректер моделі, оны сақтау оңай, деректердегі белгісіздіктер мен деректерді қайталау жоқ.

Қалыптандырудың үш деңгейінің ресми анықтамасынан кейін біз нақты мысалды талдап, мүмкін мәселелерді сипаттаймыз. Мысал ретінде студенттер қатысқан курстар туралы мәліметтер бар мәліметтер базасы қарастырылады.

Қалыпқа келтіруге дейінгі мәліметтер базасының кестелері

Бұл мысал мынаны болжайды:

- * студент курстардың кез келген санына жазыла алады;
- * оқытушылар бірнеше курстар жүргізе алады;
- * әр оқытушы әрдайым бір аудиторияда сабақ өткізеді;
- * әр аудиторияда бір ғана курс оқылады.

Осы ақпаратты сақтау үшін келесі кестелер пайдаланылсын.

Students (студенты)

Таблица 6

Name (имя)	Phoneno (телефон)	CourseRegistrations (посещаемые курсы)
Maijorie Green	415986	Basic Computing, Database Administration
Bun Gringelsby	707938	Database Administration, Advanced Hardware Support
Anico Yokamoto	415935	Advanced Hardware Support

Courses (курсы)

Таблица 7

Course (курс)	Lecturer (лектор)	Room (аудитория)
Basic Computing	Meander Smith	542 South

Database Administration	Dean Straight	221 East
Advanced Hardware Support	Dean Straight	221 East

Бұл жағдайда келесі логикалық қайшылықтар пайда болады:

* Егер негізгі есептеу курсы жабылса, кестеден Meander Smith дәріскері және 542 South аудиториясы жойылады;

* студент жазыла алатын курстардың саны Course Registrations өрісі рұқсат ететін жазу ұзындығымен шектеледі;

* Course Registrations өрісінде мәндерді іздеу, сонымен қатар оны есептеулерде пайдалану қиын;

* әр тіркеу жазбасында курстың толық атауы қайталанарды. Нәтижесінде кеңістік тиімсіз пайдаланылады және егер курстың атауы қателіктермен енгізілсе, сәйкес келмейтін деректердің пайда болу ықтималдығы артады. Сонымен қатар, курс атауын өзгерткен кезде барлық тіркеу жазбаларын іздеу және жаңарту қажет;

* студенттер кестесін тегі бойынша индекстеу мүмкін емес, өйткені name өрісі студенттердің толық аттарын сақтайды;

* егер дәріскер аудиторияны алмастырса, онда ол оқытатын барлық курстар туралы ақпаратты жаңартуға тура келеді.

Біз қалыпқа келтіреміз

Қалыпқа келтіруден кейінгі мәліметтер базасының кестелері Students (студенты)

Таблица 8

StudentsID(код студента)	Firstname (имя)	Lastname (фамилия)	Phoneno (те лефон)
1001	Maijorie	Green	415986
1002	Bun	Gringelsby	707938
1003	Anico	Yokamoto	415935

Регистрационные записи (Registrations)

Таблица 9

RegID (код записи)	StudentsID(код студента)	Courses (курсы)
1	1001	1
2	1002	2
3	1003	3
4	1004	4
5	1005	5

Courses (курсы)

Таблица 10

Course ID(курс)	Course (курс)	LecturerID (код лектора)
1	Basic Computing	1
2	Database Administration	2
3	Advanced Hardware Support	3

Lecturers (лектор)

Таблица 11

LecturerID (код лектора)	Firstname (имя)	Lastname (фамилия)	Room (аудитория)
1	Meander	Smith	542 South
2	Dean	Straight	221 East

Кестелер арасында келесі байланыстар бар:

Студенттер (Студенттер) — курстар (курстар): Registrations аралық кестесі арқылы "көпке" қатынасы (тіркеу жазбалары), басқаша айтқанда, бұл қатынас "көпке" деген екі қарым-қатынасқа азаяды;

Студенттер (Студенттер) — тіркелімдер (тіркеу жазбалары): бір-көп қарым-қатынас;

Курстар — курстар) - тіркеулер(тіркеу жазбалары): бір-көпке қатынасы;

Lecturers — лекторлар) - курстар (курстар): "бір-көп" қатынасы.

Осы кестелерді қалыпқа келтірудің айқын артықшылықтары:

- * әр кестеде байланысқан мәліметтердің бір ғана жиынтығы бар. Мысалы, студенттер кестесінде енді қатысқан курстар туралы ақпарат жоқ;

- әр кестеде бастапқы кілт бар: Студенттер кестесінде — StudentID өрісі, Registrations кестесінде-RegID, Courses кестесінде-CourseID және Lecturers кестесінде-LecturerW,

- * құрама өрістер жоқ. Әр өрісте тек бір атрибут сипатталады. Мысалы, студенттің аты-жөні көрсетілген өріс студенттің аты-жөнін қамтитын жеке өрістерге бөлінеді;

- * қайталанатын деректер жоқ. Сонымен, қазір оқытушылардың есімдері тек бір рет жазылады;

- * бірнеше мәннен тұратын өрістер жоқ. Мысалы, курстың әр тіркеуі енді Registrations кестесінің бөлек жолында орналасқан. Салыстыру үшін, студенттер кестесінің алдыңғы нұсқасының Course Registrations (курстар) өрісін қараңыз;

- * әр өріс толығымен бастапқы кілтке байланысты. Мысалы, Courses кестесінде бөлме өрісі жоқ. Себебі, аудитория курс кодына (CourseID) емес, оқытушы кодына (LecturerID) байланысты.

Қалыпқа келтірудің негізгі артықшылықтары:

- * индексті сұрыптау және құру оңай, өйткені кестелер ықшам бола бастады;

- * индекстер ықшам бола бастайды;
- * бір кестедегі индекстердің аз саны жазба жаңартуларын тезірек орындауға мүмкіндік береді;
- * кестелерде NULL мәндері мен артық деректер аз, бұл мәліметтер базасының ықшамдылығын арттырады;
- * кестелік блоктау қақтығыстарының ықтималдығы төмендейді, өйткені шектеулі мәліметтер жиынтығы бұғатталуы керек.

Реляциялық деректер базасының жобасы-бұл барлық атрибуттар анықталған, қатынастардың бастапқы кілттері берілген және тұтастықты қолдау принциптеріне қатысты қатынастардың басқа да қосымша қасиеттері берілген өзара байланысты қатынастар жиынтығы. Шын мәнінде, мәліметтер базасының жобасы - бұл ұзақ уақыт бойы және көптеген пайдаланушылар қолданатын болашақ бағдарламалық жасақтама негізі. Деректер базасының өмірлік циклінің кезеңдері (1-суретті қараңыз) негізінен кез-келген бағдарламалық жүйенің дамуына ұқсас, бірақ оларда тек мәліметтер базасына қатысты белгілі бір ерекшелік бар.



Рис. 1. Этапы жизненного цикла БД

Мәліметтер базасын жобалау процесі-бұл белгілі бір модель тұрғысынан пәндік аймақтың ақпараттық құрылымының бейресми ауызша сипаттамасынан пәндік аймақ объектілерінің формальды сипаттамасына ауысулар тізбегі. Жобалаудың келесі кезеңдерін бөлуге болады:

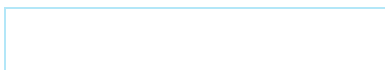
1. Пәндік аймақтың ақпараттық объектілерін жүйелі талдау және ауызша сипаттау.

2. Пәндік аймақтың инфологиялық моделін жобалау-кейбір семантикалық модель тұрғысынан, мысалы, ег моделі тұрғысынан пәндік аймақтың объектілерін ішінара формальды сипаттау.

3. Деректер базасын даталогиялық немесе логикалық жобалау, яғни мәліметтер базасын логикалық деректер моделінің қабылданған күні тұрғысынан сипаттау.

4. Деректер базасын физикалық жобалау, яғни қосымшаның ең тиімді жұмысын қамтамасыз ету үшін Дерекқорды сыртқы медиаға тиімді орналастыруды таңдау.

Егер екінші және үшінші кезеңдер арасында біздің жобамыз қандай стандартты ДҚБЖ-ны қолдану туралы шешім қабылдау қажет екенін ескеретін болсақ, онда шартты түрде мәліметтер базасын жобалау процесі бес тиісті кезеңнің реттілігімен ұсынылуы мүмкін (суретті қараңыз. 2.)



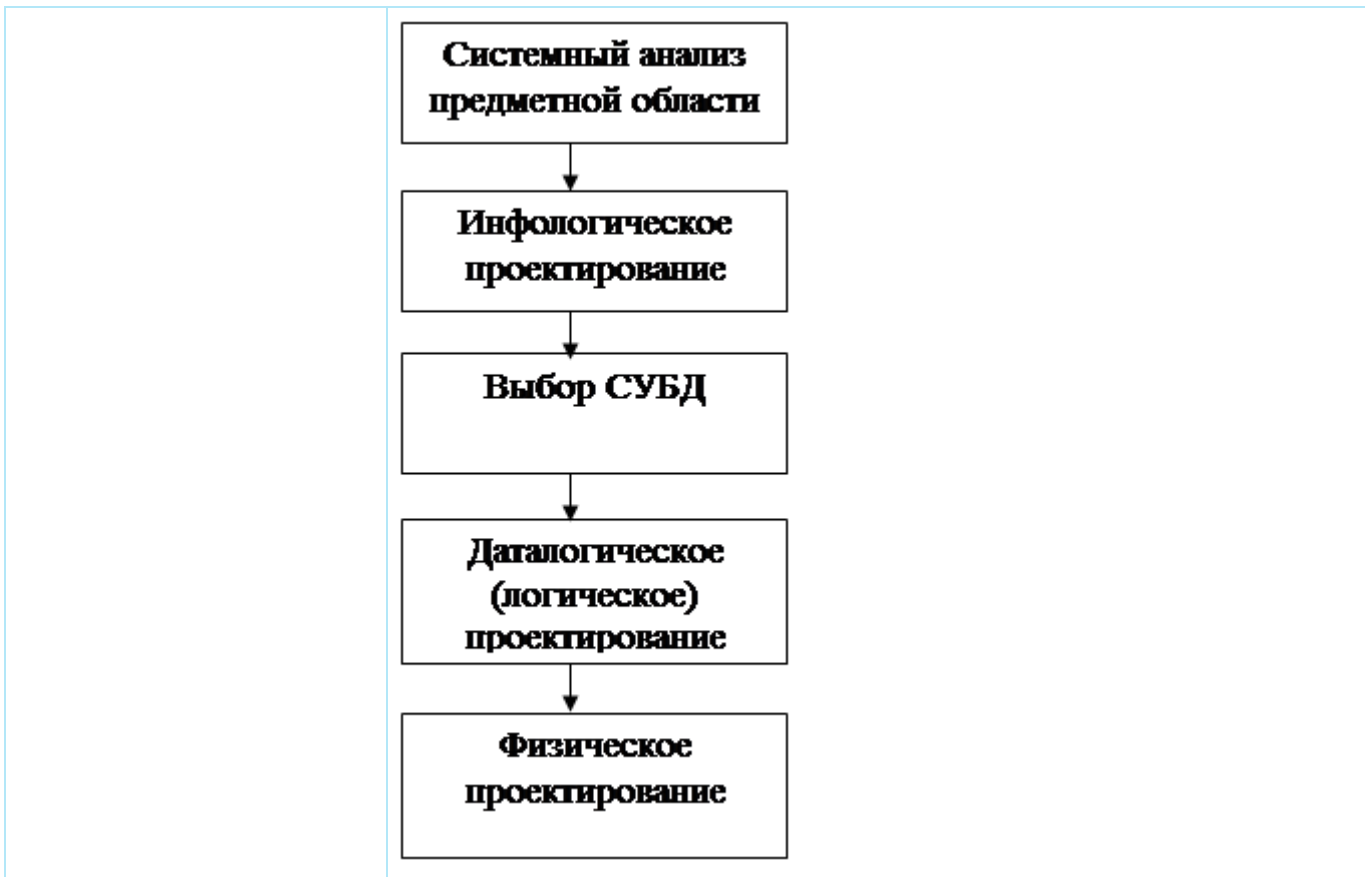


Рис. 2. Этапы проектирования БД