

Ақпаратты жылдам табу үшін деректер базасындағы жазбаларды сұрыптап алған жөн. Жазбаларды өрістері бойынша өсу немесе кему ретінде немесе алфавиттік ретте сұрыптап, тек содан ғана өздеріңе қажет категориялар бойынша ақпарат алуларыңа болады. Бұл процесс деректербазасын жобалау деп аталады.

Жобалау компьютердің көмегімен жасалады. Жобалау - деректер базасын теориялық түрде құру. Ақпараттың өте үлкен көлемін өңдеу және іздеуге арналған ДББЖ деп аталатын арнайы программалар болады.

Мәліметтер қоры – бұл ең алдымен кестелер жиынтығы белгілі бір фирманың не оқу орынның және т.б. салалардың жұмысын автоматтандыруға арналған бағдарлама болып табылады.

Мәліметтер қорын жасау үшін ең алдымен бізге кесте міндетті түрде керек. Сондықтан да біз Microsoft Office стандартты бағдарламасынан MS Access те жұмыс жасаймыз. Алдымен MS Access-ті іске қосамыз.

“Деректер қорын құру” терезесінен “Жаңа деректер қоры” ауыстырып-қосқышын таңдауымыз қажет. “Жаңа деректер қоры файлы” терезесінде деректер қоры файлы орналасатын файлды көрсетіп, файл аты өрісіне файлдың атын енгіземіз. ОК батырмасын басып кесте құруды бастаймыз. “Деректер қоры” терезесінде “Кесте” батырмасын таңдап, “Құру” батырмасын басамыз. “Жаңа кесте” терезесінде жаңа кестенің құрылымын құру режимі – Конструктор режимін таңдаймыз.

Кестелер арасындағы байланыс бір немесе бірнеше сәйкес өрістер арқылы жүзеге асырылады. Ендеше, деректер қорының келесі бөлігін Delphi ортасында шақырамыз.

Бізге керекті компоненттер бұл жерде ADO (Active Data Objects), Data Access, Data Controls бұл компоненттер бізге базамен кестемізді байланыстыратын бірден бір көпір болып табылады. ADO (Active Data Objects) технологиясының қолдануымен деректерді басқаруға арналған ADO бетіндегі компоненттері яғни бұлар: ADOConnection – Байланыс, яғни біздің қолданып отырған деректер қорымызда бұл компонент кесте жол ашушы болып табылады; ADOTable - Table Деректер жиыны, ал бұл компонентте біз Connection қасиетіне ADOConnection1 байланысын орнатып TableName қасиетінен керек кестемізді таңдаймыз; ADOQuery - Query Деректер жиыны бұл компонент құралдар тақтасындағы SQL элементі арқылы жүзеге асырылады, бұл компонентте ADOTable компоненті сияқты қызмет атқарады. Бірақта екі компонентті бірдей бір формада қолдануға болмайды екеуінің біреуі ғана қолдану керек. Ал біздің жағдайымыз ADOTable компоненті қолданылады. Ал формаға орнатылған төмендегі компоненттердің негізгі іс-әрекеттерін қарастыратын болсақ:

TTable(Кесте) - кез келген Деректер қорындағы яғни Access-тегі кестемізге қол жеткізу;

TDataSource(Берілгендер көзі) - DataSet(Берілгендер жиынтығы) қасиеті арқылы нақты кестемен байланыс орнату болып табылады;

TDBGrid(Берілгендер кестесі) - DataSource қасиетіне DataSource берілгендер көзін меншіктеп, терезесіне нақты берілгендерді (кестені не сұранысты) енгізу. Active қасиетіндегі (False немесе True) яғни ақиқат немесе жалған пунктін өзгертміз.

Ал DataSource1 компоненті өзінің DataSet (Мәліметтер жиынтығы) қасиеті арқылы нақты кестемен байланысады. Ол – Table1 және DBGrid1 компоненттерін байланыстырушысы.

Енді біздің базамен Delphi арасындағы байланыс толығымен орнатылды. Сонымен қатар базаға бірнеше батырмалар мен формдар орнатып деректер қорын ұлғайтуға болады.

Әдебиеттер тізімі

1. Мұхамбетжанова С.Т., Мелдебекова М.Т. Педагогтардың ақпараттық – коммуникациялық технологияларды қолдану бойынша құзырлықтарын қалыптастыру әдістемесі. Алматы: ЖШС «Дайыр Баспа», 2010 ж.

2. Шумаков П.В. Delphi 3 и разработка приложений баз данных; М.: Нолидж, 2010. - 704 с.

ДИАГРАММАЛАРДЫ RATIONAL ROSE-ДЕ ҚОЛДАНУДЫҢ ТИІМДІЛІГІ

Султанова Г.А., Бейсенбек А.Б.

академик Е.А. Бөкетов атындағы Қарағанды мемлекеттік университеті, Қазақстан

E-mail: gasultanova@mail.ru

Қоғамда қазіргі заман дәуіріндегі ғылыми-техникалық үрдістің дамуына байланысты адам іс-әрекетінің барлық қызметінде компьютерлік технологияны пайдалану кеңінен етек жайды. Оқу-тәрбие үрдісінің тиімділігін жоғарылатуда жаңа технологияларды пайдалану мүмкіндіктерін зерттеуге әкеледі. Білімнің дамуы барысында көптеген перспективті ғылыми жолдардың ішінде

модельдеу пайда болады. Модельдеу мәселені шешуде, құрастыруда және оларды өндірісте қолдануда кеңінен пайдаланылады. Модельдеуді қарастыру көбінесе UML тілінде жазылған, Rational Rose программасында әртүрлі диаграммалар арқылы іске асады. Rational Rose UML тіліне негізделіп жобалау және объектілі-бағытталған талдау әдістерін қолданады.

Rational Rose жаңа жобаларда бағдарламалық компоненттерінің қайта қолдануын қамтамасыз ететін бағдарламалар мен мәліметтер қорының реверстік инжинирингтің құралдарынан тұрады. Мәселені шешу аумағының концептуалды моделін құру Rational Rose программасында case құралы арқылы іске асырылды. Rational Rose бағдарламасы ақпараттық жүйелерді, бағдарламаларды жобалауда кеңінен қолданылатын құрал case болып табылады. Ол объектіге-бағытталған принциппен жұмыс істейді. Графиктік модельдер UML тілінің көмегімен құрылады. UML тілінің көмегімен құрылған графиктік модельдер диаграммалардан тұрады. Rational Rose-де жұмыс істеудің негізі жүйе архитектурасының статистикалық және динамикалық аспектілерін анықтайтын UML егжей-тегжейімен диаграммаларды құру болып табылады[1].

Диаграммалар түгелдей объектіге-бағытталған әдістердің негізгі қалаушысы болып табылады. Объектіге-бағытталған бағдарламалар жүйесінде модельдеуді қарастыруда диаграммалар маңызды орын алады. Диаграмма жүйедегі объектілердің типін анықтау және олардың арасындағы байланысты көрсетеді. Диаграммасы дербес жағдайда заттық облыстың арасындағы, яғни объектілер мен ішкі жүйелер сияқты өзара қарым-қатынасты сипаттайды. Сонымен қатар олардың ішкі құрылымын және қатынас түрлерін қарастырады. Диаграммаларды көбінесе модельдеу кезінде қолданады, олар жүйенің құрылымын көрсете отырып жобалауды сипаттайды. Диаграммалар кластарда суреттелген объектілердің үдемелі тәртібін көрсетпейтіндіктен, кластар мен интерфейстер арасындағы қатынастарды көрсетеді.

Диаграммалар әртүрлі түрлерімен байланысқан құрылымдық қатынастарда «классификатор» типті элементтер секілді ең жоғарғы түсінік болып саналады. Диаграммалар пакеттер, қатынастар, типті объектілер және байланыстар сияқты жеке дана ұғымдарды іске асырады. Диаграмманы құрудан бұрын, ол диаграмманың қандай мақсатта қолданылатынын анықтап алу керек[3]. Rational Rose программасында диаграммалардың 13 типі сипатталады. Бұл диаграммалар көптеген мамандарға әртүрлі амалдарымен көрсетіледі. Бірақ, мен бұл диаграммаларды тілдерді құрастыратын орта деп ойламаймын, өйткені олар әлі де толығымен анықталмаған. Көбнесе бір типтегі диаграмма элементтері, басқа диаграммаларда болуына мүмкіндігі бар. UML стандартында, әдетте белгілі элементтер, типіне сәйкес келетін диаграммаларда сызылады. Жақсы диаграммалар көбінесе, жоба туралы идеяларды алмасуда қолданылады. Сонымен қатар диаграммалар программалық жүйені және бизнес-жоспарларды жан-жақты қамтамасыз етуге көмегін тигізеді. Кейде, жұмыс тобы кейбір жұмыстың мәселесін ашуға тырысып жатқанда, диаграммалар өзара түсінушілікті орнатуға және кеңінен таралуына септігін тигізеді[2].

Диаграммаларға байланысты ең үлкен қауіп, мәселені игеруде өте ерте шешім қабылдауға тура келеді. Бұған қарсы тұру үшін концептуалды аспектіге және аспектінің спецификациясына көңіл бөлген дұрыс. Диаграммалар заманауи модельдеу құжаттарының маңыздылығын арттырады. Олар күрделі бағдарламалық концепцияға деген түсінікті күшейтеді, сонымен қатар мәселені талдауға және шешімді жобалауға көмектеседі. Өз кезеңінде диаграммалар жұмыстың мәнін айтарлықтай жоғарылатады.

Әдебиеттер тізімі

1. Сатыбалдиева Р.Ж. UML-ді объектіге бағытталған талдауда және жобалауда қолдану. –М.: Алматы, 2014. «Print S» баспасы -Б.32-35
2. Рамбо Дж. UML 2,0 объектно-ориентированное моделирование и разработка. 2-е изд.-СПб.:Питер, 2007. –С.42-76.
3. Арлоу Д. UML 2 и унифицированный процесс. Практический объектно-ориентированный анализ и проектирование. 2-е изд. Пер. с англ. –Спб.: Символ Плюс, 2014. –С.147-174.