**Алтай Шалтабаев,**

**Гүлдана Алмасқызы, Лаура Құрман**

**(Талдыкорган, Республика Казахстан)**

**Визуалды компоненттер кітапханасы**

Егеменді тәуелсіз тұғыры биік еліміздің әлеуметтік-экономикалық жағдайы ақпараттық технологияны өмірдің әрбір саласында жаппай пайдаланумен сипатталады. Студенттердің танымдық іс-әрекеттері күшейіп, өзіндік жұмыстарды тез орындау мүмкіндіктері артты.

Бүгінгі таңда мәліметтерді сақтау және өңдеу жұмыстарын соңғы жылдарда шыққан программалық өнімдер арқылы жылдам әрі тиімді ұйымдастыруға болады. Яғни, біз визуалды компоненттер кітапханасы жайлыақпарат берейік.

Визуалды компоненттер кітапханасы (Visual Component Library, VCL) қолданбаны тез өңдеуге арналған көптеген кластардан тұрады. Кітапхана Object Pascal тілінде жазылған және көптеген мүмкіндіктері бар Delphі-дің қолданбалар құрастыру ортасымен тікелей байланысты. Визуалды компоненттер кітапханасы деп аталғанмен мұның құрамына визуалды емес компоненттер де кіреді.

Алайда, визуалды компоненттермен бірге ТОbject абстрактілі класынан бастап басқа да, кластар бар. Барлық компоненттер класс болып табылады, бірақ барлық кластар компонент бола бермейді. VCL-дің барлық кластары клатар тармағын құрайды және олар тармақтың белгілі бір деңгейінде орналасқан. Объектің шығу тегін білу оны оқып-үйренуде көп көмегін тигізеді. Өйткені, ұрпақ ата-ана объектісінің барлық элементтеріне ие. Мысалы, Caption қасиеті TControl класына жататын болса,онда бұл қасиет оның ұрпақтарында да болады.

Барлық Object Pascal кластардың ортақ ата-анасы тармақ түбінде орналасқан ТОbject класы болып табылады. Бұл класс абстрактілі болып табылады және барлық ұрпақ кластарға ортақ әдістерді таратады.

Негізгі әдістер:

* Create (объектіні құру)
* Destroy (объектіні өшіру)
* Free (Create әдісімен құрылған объектіні өшіреді, сонымен қатар бұл кезде Destroy әдісі де шақырылады).

Бұл әдістердің көбісі ұрпақ кластарда қайта анықталады.

TРersistent, TСomponent және TСontrol ұрпақтарына сипаттама берейік.

TРersistent класы қасиеттері ағыннан іске қосылатын және сонда сақталатын объектілер үшін абстрактілі класс болып табылады. ТОbject класының әдістеріне қосымша, TРersistent бір объектінің өрістерімен қасиеттерін екіншісіне беруге мүмкіндік беретін Assign әдісіне ие.

Tсomponent класы барлық компоненттер үшін базалық болып табылады және өзінің ата-анасының әдістеріне қосымша компоненттер басқа компоненттерді иеленетін құралдарды ұсыныды. Нәтижесінде формаға кез-келген компонентті қойған кезде, ол басқа компонентке жататын болады. Компонентті құрған кезде ол компонентке тиісті барлық компоненттердің автоматты түрде құрылуын қамтамасыз етеді, ал ол компонентті өшірген кезде оған тиісті компоненттің барлығы автоматты түрде өшіріледі. TСomponent класының кейбір қасиеттері мен әдістерін атап өтейік.

*Қасиеттері:*

* Components (оған тиісті компоненттер тізімі)
* CopponentCount(тиісті компоненттер саны)
* Copponentindex(тиісті компоненттер тізіміндегі компонент нөмірі)
* CopponentState (ағымдағы компоненттің күйі)
* Name (компонент аты)
* Owner (компонент иесі)
* Tag (компонентпен бірге сақталатын бүтін мән).

*Әдістер:*

* DestroyComponent (барлық тиісті компоненттерді бұзады)
* Destroying (тиісті компонентке оның бұзылытындығы жайлы айтады)
* FindComponent (Components тізімінен компонентті табады.

TСomponent класынан визуалды және визуалды емес компоненттер пайда болады.

TСontrol класы визуалды компоненттердің базалық класы болып табылады және олардың қызмет етуіне негізгі құрылғыларды қамтамасыз етеді.

Барлық визуалды компоненттер терезелік және графикалық болып бөлінеді.

TWinControl және TGrapihcControl класынан шығады.

Визуалды компоненттердің VCL кітапханасында TControl класы барлық компоненттер үшін базалық класс болып табылады. Ол элементтің орны мен өлшемі, тақырыбы, түсі және тағы да басқа параметрлары сияқты негізгі функционалдық атрибуттарды қамтамасыз етеді. TControl класы визуалды компонеттерге ортақ қасиеттерден, оқиғалардан және әдістерден тұрады. Визуалды компоненттерді екі үлкен топқа бөлуге болады: басқарудың терезелік және терезелік емес элементтері [3].

*Басқарудың терезелік элементі* – нақты бір мақсатқа арналған арнайы терезе. Бұндай терезелерге мысалы, батырмалар, мәтіндік өрістер, айналу сызықтары жатады. Басқарудың терезелік элементтері үшін базалық класс – TControl класының тікелей ұрпағы TWinControl болып табылады.

Енгізу фокусын алу үшін басқарудың терезелік элементтері екі әдіспен әсер етуі мүмкін:

* редакторлеу курсорының көмегімен;
* фокус тіктөртбұрышының көмегімен.

Edit, DbEdit, Memo, DbMemo сияқты редакторлар енгізу фокусын алған кезде өз облысында редакторлеу курсорын бейнелейді. Үндемеген жағдайда редакторлеу курсоры жыпылықтаған тік тік сызық түрінде болады және пернетақтадан енгізілетін символдың ағымдық орнын көрсетеді. Редакторлеу курсоры курсорды басқару батырмаларының көмегімен жылжып отырады.

Мәтінді редакторлеумен байланысы жоқ компоненттерде, әдетте енгізу фокусын алу үшін пунктирлі қара тіктөртбұрыш пайдаланылады. Мысалы, Button батырмасы үшін бұл тіктөрбұрыш тақырыбының айналасында пайда болады, ал ListBox және DbListBox тізімдері үшін тіктөртбұрыш ағымдағы уақытта таңдалынған жолды белгілейді. Таңдалынған жол басқа түске боялуы мүмкін.

Басқарудың терезелік емес элементтері үшін TСontrol класынан өңделген TGraphicControl класы базалық класс болып табылады. Басқарудың терезелік емес элементтерінде енгізу фокусы болмайды және олар басқа элементтердің ата-анасы бола алмайды. Басқарудың терезелік емес элементтерін басқарудың терезелік элементтерімен салыстырғандағы артықшылығы қорлардың аз шығындалуы болып табылады. Басқарудың терезелік емес элементтеріне Label, DBText т.б. компоненттер жатады.

Программаны орындау барысында компонент құру мысалын қарастырайық:

procedure TForm1.Button1Click(Sender: TObject);

begin

With TEdit.Create(Self) do

begin

Parent:=Form1;

Name:='edtName';

text:='К.Р.Досымбеков';

Left:=100;

Top:=60;

end;

end;

end.

Бұл жерде Button1 батырмасын шерткен кезде Tedit типіндегі компонент динамикалық түрде құрылады. Егер Parent қасиетін көрсетпесек онда компонент құрылады, бірақ экранда көрінбейді. Name қасиетін көрсетпесек компонентке Edit1 аты меншіктелінеді. Тext қасиетін көрсетпесек ол Name қасиетінің мәніне тең болады. Сол сияқты Visible қасиетін көрсетуге болады.

*Визуалды компоненттер.* Қолданбаның интерфейсін құру үшін Delphi жүйесі визуалды компоненттердің көптеген жиынтығын ұсынады. Олардың негізгілері компоненттер палитрасының мына беттерінде орналасқан: *Standard, Additional* және *Win32.*

Компоненттерді бұлай бөлу олардың атқаратын қызметтерімен мүмкіндіктеріне қарай емес, парақтар атауына байланысты болып отырған сияқты. Мысалы Button және BitBtn батырмалары әртүрлі парақтарда орналасқандығымен, іс-жүзінде атқаратын қызметтері бірдей. Парақтардағы бірінші шартбелгі компонентке жатпайды. Ол парақтан компонент таңдауды қайтаруқұралы. Компоненттер екінші шартбелгіден басталады.

Компоненттер Палитрасының Standard парағы 1-кестеде көрсетілген.

*Standard парақ.* Компоненттер Палитрасының Standard парағында көбі Windows-тың алғашқы нұсқаларында пайдаланылған, интерфейсті компоненттер орналасқан. Олар 1-кестеде көрсетілген.

1-кесте.*Standardпарағыныңкомпоненттері*

|  |  |
| --- | --- |
| *Аты* | *Қызметі* |
| *Frame* | *Рама. Формаменқатарбасқакомпоненттердіорналастыруғаарналған контейнер. ФормаданайырмашылығыолКомпоненттерпалитрасындаорналасады.* |
| *MainMenu* | *Проектінің бас менюі. Компонент күрделі сатылы менюларды құрып, олардың қызметін іске асыра алады.* |

1-кестенің жалғасы.*Standardпарағыныңкомпоненттері*

|  |  |
| --- | --- |
| *Button* | *Командалық батырма. Бұл компоненттің OnClick оқиғасын өңдеуіш, әдетте, кейбір командаларды орындау үшін пайдаланылады.* |
| *PopupMenu* | *Көмекші немесе локальді меню. Әдетте бұл меню тышқанның оң жақ батырмасын шерткенде жеке терезеде пайда болады.* |
| *CheckBox* | *Тәуелсіз ауыстырыпқосқыш. Жұмыс істеп тұрған программада бұл компонентті шерту оның Checked (таңдалған, таңдалмаған) деген логикалық қасиетін өзгертеді.* |
| *RadioButton* | *Тәуелді ауыстырыпқосқыш. Әдетте топқа кемінде осындай бір компонентпен біріктіріледі. Ауыстырыпқосқышты шерту осы топтағы алдыңғы таңдалынған ауыстырыпқосқышты автоматты түрде босатуға алып келеді.* |
| *ListBox* | *Таңдау тізімі. Таңдалатын тізімнен тұрады және таңдалған тізім элементін бақылауға мүмкіндік береді.* |
| *ComboBox* | *Таңдаудың аралас тізімі. Жолдар тізімін ашылған немесе ашылатын тізім түрінде бейнелейді.* |
| *ScrollBar* | *Жылжыту жолағы. Ол тік және көлденең жолақ түрінде болады.* |
| *GroupBox* | *Элементтер тобы. Бұл компонент мағынсы бойынша бірнеше компоненттерді топтастыру үшін пайдаланылады.* |
| *RadioGroup* | *Тәуелді ауыстырыпқосқыштар тобы. Жүйелі түрде орналасқан RadioButton батырмалар тобын құруда пайдаланылады.* |
| *Panel* | *Панель. Бұл компонент GroupBox компоненті тәрізді бірнеше компоненттерді біріктіру үшін пайдаланылады. Мәтінді шығару компоненті ретінде де пайдаланыла алады.* |
| *ActionList* | *Әрекеттер тізімі.* |
| *Label* | *Белгі. Бұл компонент терезеде шағын жазбалардыорналастыруғапайдаланылады.* |
| *Edit* | *Біржолды мәтіндік редактор. Бір жолдық мәтінді шығаруға, енгізуге немесе редактірлеуге арналған.* |
| *Memo* | *Көпжолды мәтіндік редактор. Көпжолды мәтінді енгізуде және/немесе бейнелеуде пайдаланылады.* |

*Additional парақ.* Additional парағында диалогтық терезелердің түрлерін әртүрлі етіп жасауға мүмкіндік беретін қосымша компоненттерін кеесі мақалада жариялаймыз.

**Қолданылған әдебиеттер тізімі:**

1. ГофманВ., Хомоненко А, “Delphi 7” Санк-Петербург 2003 г.
2. Н.Культин. «Основы программирования в Delphi 7». «БХВ-Петербург». 2003г.
3. А. Я. Архангельский. Програмирование в Delphi 6. М: “Бином”, 2002.
4. Баженова И. Ю. Delphi 7 самоучитель програмиста. Учебно – справочное пособье. М: “ Кудиц - Образ ”, 2003.

**Научный руководитель:**

(Ph.D.) –Технических наук Шалтабаев Алтай Аканович.