

УДК 004.5 + 004.512.3+004.4

РАЗРАБОТКА ИНТЕРФЕЙСА ПОИСКОВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ С ИСПОЛЬЗОВАНИЕМ BORLAND

А.В. Выборщик

В статье рассмотрена разработка интерфейса программного обеспечения, позволяющего получить выходные данные в таблицах по известным различным наборам исходных данных. Результаты исследований используются для расчёта нормативов режимов резания.

Ключевые слова: объектно-ориентированное программирование, база данных, интерфейс, программное обеспечение.

Одной из задач, стоящих перед разработчиками современного программного обеспечения, является разработка интерфейса, который, с одной стороны, должен отвечать требованию максимальной информативности, а с другой – быть максимально доступен. Основными недостатками интерфейса программ, разрабатываемых в том числе и для решения инженерных нужд, являются, во-первых, неудобство доступа ко многим функциям, во-вторых, перегруженность различного рода компонентами, не несущими информационной нагрузки, в-третьих, недостаточные по размеру окна для ввода и вывода, в-четвёртых, отсутствие защиты от ввода некорректных или бессмысленных данных, в-пятых, неудобное расположение самих окон ввода и вывода. Перечисленные недостатки значительно снижают скорость работы с программным обеспечением, что ведёт к большим временным затратам. Необходимость постоянно отвлекаться на учёт указанных недостатков в итоге приводит к отказу от использования программного обеспечения.

Поэтому для широкого и производительного использования программного обеспечения необходима не только корректная работа самих программ, но и интерфейс, являющийся удобным средством работы с компьютером.

При разработке интерфейса необходимо выполнение следующих действий: определение минимально необходимого количества окон; корректное открытие и закрытие окон; корректное отображение компонентов ввода и вывода данных, а при большом количестве исходных данных – с группировкой по типам данных; желательное указание пользователю на последовательность ввода исходных данных; помещение компонентов, используемых для подтверждения ввода исходных данных; использование компонентов для необходимых пояснений.

Как правило, количество окон в приложении определяется количеством баз данных, для которых составляется; чаще всего таких окон три: главное

окно, окно расчёта и окно информации о программе, разработчике и т.п. Открытие и закрытие окон выполняется с использованием событий OnActivate, OnCreate, OnClose. Для корректного последовательного ввода данных удобнее всего использовать фокус ввода, а для избежания нарушения последовательности ввода – включение/выключение опции ввода данных. Для группировки исходных данных и конечного результата, как правило, используются декоративные компоненты, например, GroupBox. Необходимые пояснения можно выполнить использованием свойства Hint или помещением отдельных компонентов, например, Label.

Пример фрагмента программы, формирующего интерфейс, приведён ниже.

```
procedure TForm2.Button1Click(Sender: TObject);
var kf1,kf2,kf3,kf4,kf5,kf6,rf1,rf2,rf3,rf4,rf5,rf6: string;
    kv1,kv2,kv3,kv4,kv5,kv6: variant;
    vrnres1,vrnres2,vrnres3,vrnres4,vrnres5,vrnres6: variant;
    strres1,strres2,strres3,strres4,strres5,strres6: string;
    m1,m2,m6: integer;
    m3,m40,m41: real;
    label 1, 2, 3, 4, 5, 6, 7, 8, 9, 10;
begin
    kf1:='Mat_mach';
    kv1:=dbcombobox1.text;
    rf1:='Code_mat_mach';
    vrnres1:=table1.lookup(kf1,kv1,rf1);
    m1:=integer(vrnres1);
    1: table2.First;
    repeat
    2: if m1=table2.FieldName('Code_mat_mach').AsInteger then
    3: if strtfloat(trim(maskedit1.text))<=table2.FieldName('HB').Asfloat
    then
    begin
    kf2:='Code_mat_mach;HB';
    kv2:=vararrayof([table2.FieldName('Code_mat_mach').AsInteger,
    table2.FieldName('HB').Asfloat]);
    rf2:='Code_HB';
    vrnres2:=table2.lookup(kf2,kv2,rf2);
    m2:=integer(vrnres2);
    goto 4;
    end
    else
    begin
    table2.Next;
```

```
goto 3;
end
else
begin
table2.Next;
goto 2;
end;
until table2.Eof;
4: table3.First;
repeat
5: if m2=table3.FieldName('Code_HB').AsInteger then
6: if strtofloat(trim(maskedit2.text))<=table3.FieldName('d_1').Asfloat
then
begin
kf3:='Code_HB;d_1';
kv3:=varrayof([table3.FieldName('Code_HB').AsInteger,
table3.FieldName('d_1').Asfloat]);
rf3:='v';
vrnres3:=table3.lookup(kf3,kv3,rf3);
m3:=real(vrnres3);
goto 7;
end
table3.Next;
else
begin
goto 6;
else
end
begin
table3.Next;
goto 5;
end;
until table3.Eof;
7: maskedit3.Text:=floattostr(m3);
table4.First;
kf4:='Code_HB;d_2';
rf4:='S;n';
repeat
8: if m2=table4.FieldName('Code_HB').AsInteger then
9: if strtofloat(trim(maskedit2.text))<=table4.FieldName('d_2').Asfloat
then
begin
```

```
kv4:=vararrayof([m2, table4.FieldName('d_2').Asfloat]);
vrnres4:=table4.lookup(kf4,kv4,rf4);
m40:=real(vrnres4[0]);
m41:=real(vrnres4[1]);
goto 10;
end
else
begin
table4.Next;
goto 9;
end
else begin
table4.Next;
goto 8;
end;
until table4.Eof;
10: maskedit4.Text:=floattostr(m40);
maskedit5.Text:=floattostr(m41);
kf6:='Code_Mat_mach';
kv6:=m1;
rf6:='Code_cool';
vrnres6:=table6.lookup(kf6,kv6,rf6);
// combobox1.items.add(string(vrnres6));
m6:=integer(vrnres6);
// table5.First;
kf5:='Code_cool';
kv5:=m6;
rf5:='Cool';
vrnres5:=table5.lookup(kf5,kv5,rf5); student
Maskedit6.Text:=String(vrnres5);
end;
```

Выполнение перечисленных требований позволяет сформировать корректный и удобный для пользователя интерфейс.

[К содержанию](#)