

**ФЕДЕРАЛЬНОЕ АГЕНТСТВО ОБРАЗОВАНИЯ
ДИМИТРОВГРАДСКИЙ ИНСТИТУТ ТЕХНОЛОГИЙ,
УПРАВЛЕНИЯ И ДИЗАЙНА**

**Лабораторная работа №4
По курсу: "Базы данных"
Вариант №3**

**Выполнил студент гр.ВТ-31:
Потеренко А.Г.
Проверил преподаватель:
Нестеркин В.Н.**

Димитровград 2005г.

Содержание

Задание.....	3
Проектирование БД.....	3
Реализация алгоритма на PL/SQL.....	4
Реализация алгоритма на DELPHI.....	9

Задание

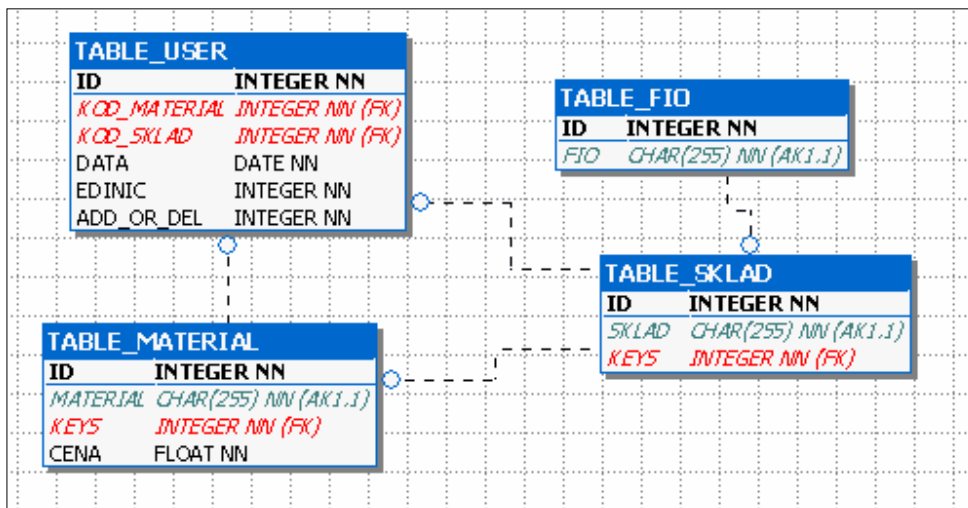
Создать базу данных, содержащую оговоренные данные. БД должна находиться минимум в третьей нормальной форме. БД должна быть целостной и непротиворечивой. Приложение должно быть построено с использованием архитектуры клиент-сервер. Физическое расположение таблиц БД – строго оговоренный рабочий каталог.

Спроектировать приложение при помощи любого языка программирования, которое позволит выводить БД на экран в читаемом виде, добавление новых записей в таблицы, редактирование и удаление с помощью SQL-запросов. Реализовать возможность вывода любой таблицы на печать при помощи стандартных компонентов – отчетов.

После выполнения работы необходимо предоставить отчет, в котором должен быть описан порядок выполнения работы с пояснениями.

Разработать приложение для складского учета. Реализовать многопользовательский режим работы с БД с разделением привилегий (администратор может заносить новую информацию в справочники, просматривать отчеты и т.д., а пользователь – лишь заносить текущие документы). В БД должно быть реализовано: справочник складов с ФИО зав.складом; справочник материалов, с указанием текущей цены, места расположения и подсчитанным количеством на основании введенных документов; в БД должны фиксироваться данные о поступлении и списании того или иного товара с выбранного склада. Реализовать возможность просмотра сведений о часто продаваемых товарах, о часто приобретаемых, состоянии дел на складах (какого товара много, какого мало), все должны просматриваться на введенную дату.

Проектирование БД



Данная таблица в полной мере находится в 3 Н.Ф.

1. Каждый атрибут атомарен.
2. Каждый неключевой атрибут функционально полно зависит от первичного ключа.
2. Нет транзитивных зависимостей между неключевыми атрибутами.

Реализация алгоритма на PL/SQL

Скрипт создания таблиц БД

```
DROP TABLE TABLE_USER;
DROP TABLE TABLE_MATERIAL;
DROP TABLE TABLE_SKLAD;
DROP TABLE TABLE_FIO;
--*****
CREATE TABLE TABLE_FIO
(
    ID INTEGER NOT NULL,
    FIO CHAR(255) NOT NULL,
    CONSTRAINT AK_2 UNIQUE (FIO)
);
ALTER TABLE TABLE_FIO
    ADD CONSTRAINT PK_3 PRIMARY KEY (ID);
--*****
CREATE TABLE TABLE_SKLAD
(
    ID INTEGER NOT NULL,
    SKLAD CHAR(255) NOT NULL,
    KEYS INTEGER NOT NULL,
    CONSTRAINT AK_1 UNIQUE (SKLAD)
);
ALTER TABLE TABLE_SKLAD
    ADD CONSTRAINT PK_2 PRIMARY KEY (ID);
ALTER TABLE TABLE_SKLAD ADD CONSTRAINT FK_1
    FOREIGN KEY (KEYS)
        REFERENCES TABLE_FIO;
--*****
CREATE TABLE TABLE_MATERIAL
(
    ID INTEGER NOT NULL,
    MATERIAL CHAR(255) NOT NULL,
    KEYS INTEGER NOT NULL,
    CENA FLOAT NOT NULL,
    CONSTRAINT AK_3 UNIQUE (MATERIAL)
);
ALTER TABLE TABLE_MATERIAL
    ADD CONSTRAINT PK_4 PRIMARY KEY (ID);
ALTER TABLE TABLE_MATERIAL ADD CONSTRAINT FK_4
    FOREIGN KEY (KEYS)
        REFERENCES TABLE_SKLAD;
--*****
CREATE TABLE TABLE_USER
(
    ID INTEGER DEFAULT 0 NOT NULL,
    KOD_MATERIAL INTEGER NOT NULL,
    KOD_SKLAD INTEGER NOT NULL,
    DATA DATE NOT NULL,
    EDINIC INTEGER NOT NULL,
    ADD_OR_DEL INTEGER NOT NULL
);
ALTER TABLE TABLE_USER
    ADD CONSTRAINT PK_1 PRIMARY KEY (ID);
ALTER TABLE TABLE_USER ADD CONSTRAINT FK_2
    FOREIGN KEY (KOD_SKLAD)
        REFERENCES TABLE_SKLAD;
ALTER TABLE TABLE_USER ADD CONSTRAINT FK_5
    FOREIGN KEY (KOD_MATERIAL)
        REFERENCES TABLE_MATERIAL;
```

Алгоритм процедуры ADD_SKLAD_FIO

Возможны 4 случая при выполнении этой процедуры:

1. Вводимая FIO совпадает с той, которая уже есть в БД. Название склада не совпадает.
2. Вводимая FIO не совпадает с той, которая уже есть в БД. Название склада не совпадает.
3. Вводимая FIO не совпадает с той, которая уже есть в БД. Название склада уже есть в БД.
4. Вводимая FIO совпадает с той, которая уже есть в БД. Название склада уже есть в БД.

Алгоритм:

1. FIO не вносится, а вставляется запись о складе. Указатель на FIO устанавливается в KEYS склада.
2. FIO заносится в БД и название склада заносится в БД. Указатель на FIO устанавливается в KEYS склада.
3. Заведующий у склада уже есть, поэтому поменять новым ЗС можно только при обновлении. Тогда и будет занесен новый ЗС в БД. Никаких действий производить не надо.
4. Никаких действий производить не надо – идет абсолютный повтор значений, введенных ранее.

Листинг процедуры:

```
CREATE OR REPLACE PROCEDURE ADD_SKLAD_FIO
(
  P_FIO TABLE_FIO.FIO%TYPE,
  P_SKLAD TABLE_SKLAD.SKLAD%TYPE
) IS
  X INTEGER;
  X1 INTEGER;
  V TABLE_FIO.FIO%TYPE;
  CURSOR C_K IS
    SELECT FIO FROM TABLE_FIO WHERE TABLE_FIO.FIO=P_FIO;
  V1 TABLE_SKLAD.SKLAD%TYPE;
  CURSOR C_K1 IS
    SELECT SKLAD FROM TABLE_SKLAD WHERE TABLE_SKLAD.SKLAD=P_SKLAD;
  I TABLE_FIO.ID%TYPE;
  CURSOR C_I IS --ПО FIO УЗНАЕМ ID
    SELECT ID FROM TABLE_FIO WHERE TABLE_FIO.FIO=P_FIO;
BEGIN
  --////////////////////////////////////
  OPEN C_K;
  FETCH C_K INTO V;
  IF V IS NULL THEN
    -----ТАКОЙ FIO НЕТ-----
    --////////////////////////////////////
    OPEN C_K1;
    FETCH C_K1 INTO V1;
    IF V1 IS NULL THEN
      -----ТАКОГО СКЛАДА НЕТ-----
      SELECT MAX(ID) INTO X1 FROM TABLE_FIO;
      IF X1>0 THEN X1:=X1+1; ELSE X1:=1;END IF;
      INSERT INTO TABLE_FIO (ID,FIO) VALUES (X1,P_FIO);
      -----
      SELECT MAX(ID) INTO X FROM TABLE_SKLAD;
      IF X>0 THEN X:=X+1; ELSE X:=1;END IF;
      INSERT INTO TABLE_SKLAD (ID,SKLAD,KEYS) VALUES (X,P_SKLAD,X1);
    ELSE
      -----ТАКОЙ СКЛАД ЕСТЬ-----
      NULL;
    END IF;
    CLOSE C_K1;
    --////////////////////////////////////
  ELSE
    -----ТАКАЯ FIO ЕСТЬ-----
    --////////////////////////////////////
```

```
OPEN C_K1;
FETCH C_K1 INTO V1;
IF V1 IS NULL THEN
    -----ТАКОГО СКЛАДА НЕТ-----
    SELECT MAX(ID) INTO X1 FROM TABLE_SKLAD;
    IF X1>0 THEN X1:=X1+1; ELSE X1:=1;END IF;
    OPEN C_I;
    FETCH C_I INTO I;
    INSERT INTO TABLE_SKLAD (ID,SKLAD,KEYS) VALUES (X1,P_SKLAD,I);
    CLOSE C_I;
ELSE
    -----ТАКОЙ СКЛАД ЕСТЬ-----
    NULL;
END IF;
CLOSE C_K1;
--////////////////////////////////////
END IF;
CLOSE C_K;
COMMIT;
--////////////////////////////////////
END ADD_SKLAD_FIO;
```

Алгоритм процедуры ADD_MATERIAL

Листинг процедуры:

```
CREATE OR REPLACE PROCEDURE ADD_MATERIAL
(
    P_MATERIAL TABLE_MATERIAL.MATERIAL%TYPE,
    P_SKLAD TABLE_SKLAD.SKLAD%TYPE,
    P_CENA TABLE_MATERIAL.CENA%TYPE
) IS
    X INTEGER;
    I INTEGER;
    D_SKLAD CHAR(255);
    D_FIO CHAR(255);
    CURSOR C_I IS
        SELECT ID FROM TABLE_SKLAD WHERE TABLE_SKLAD.SKLAD=P_SKLAD;
    CURSOR C_D IS
        SELECT KEYS FROM TABLE_SKLAD WHERE TABLE_SKLAD.SKLAD=P_SKLAD;
    CURSOR C_S IS
        SELECT FIO FROM TABLE_FIO WHERE ID=X;
BEGIN
    SELECT MAX(ID) INTO X FROM TABLE_MATERIAL;
    IF X>0 THEN X:=X+1; ELSE X:=1;END IF;
    OPEN C_I;
    FETCH C_I INTO I;
    INSERT INTO TABLE_MATERIAL (ID,MATERIAL,KEYS,CENA) VALUES (X,P_MATERIAL,I,P_CENA);
    CLOSE C_I;
    COMMIT;
END ADD_MATERIAL;
```

Дополнительные процедуры

```
CREATE OR REPLACE PROCEDURE Proc_User_Add
(
  P_MAT TABLE_MATERIAL.MATERIAL%TYPE,
  P_SKLAD TABLE_SKLAD.SKLAD%TYPE,
  EDIN INTEGER,
  DATAS DATE
)
AS
  X INTEGER;
  K1 INTEGER;
  K2 INTEGER;
BEGIN
  SELECT MAX(ID) INTO X FROM TABLE_USER;
  IF X>0 THEN X:=X+1; ELSE X:=1;END IF;
  SELECT ID INTO K1 FROM TABLE_MATERIAL WHERE MATERIAL=P_MAT;
  SELECT ID INTO K2 FROM TABLE_SKLAD WHERE SKLAD=P_SKLAD;
  INSERT INTO TABLE_USER(ID,KOD_MATERIAL,KOD_SKLAD,DATA,EDINIC,ADD_OR_DEL)
    VALUES (X,K1,K2,DATAS,EDIN,1);
END Proc_User_Add;
/
--////////////////////////////////////
CREATE OR REPLACE PROCEDURE Proc_User_Del
(
  P_MAT TABLE_MATERIAL.MATERIAL%TYPE,
  P_SKLAD TABLE_SKLAD.SKLAD%TYPE,
  EDIN INTEGER,
  DATAS DATE
)
AS
  X INTEGER;
  K1 INTEGER;
  K2 INTEGER;
BEGIN
  SELECT MAX(ID) INTO X FROM TABLE_USER;
  IF X>0 THEN X:=X+1; ELSE X:=1;END IF;
  SELECT ID INTO K1 FROM TABLE_MATERIAL WHERE MATERIAL=P_MAT;
  SELECT ID INTO K2 FROM TABLE_SKLAD WHERE SKLAD=P_SKLAD;
  INSERT INTO TABLE_USER(ID,KOD_MATERIAL,KOD_SKLAD,DATA,EDINIC,ADD_OR_DEL)
    VALUES (X,K1,K2,DATAS,EDIN,0);
END Proc_User_Del;
/
--////////////////////////////////////
CREATE OR REPLACE PROCEDURE Get_Edinic
(
  P_DATA IN TABLE_USER.DATA%TYPE,
  P_KOD IN INTEGER,
  EDIN OUT INTEGER
)
IS
  X INTEGER;
  CURSOR C_K IS
    SELECT EDINIC FROM TABLE_USER WHERE
      (DATA=P_DATA)AND(KOD_MATERIAL=P_KOD)AND(ADD_OR_DEL=1);
BEGIN
  X:=0;
  FOR V IN C_K LOOP
    X:=X+V.EDINIC;
  END LOOP;
  EDIN:=X;
END Get_Edinic;
/
--////////////////////////////////////
CREATE OR REPLACE PROCEDURE Get_Edinic_DIL
(
  P_DATA IN TABLE_USER.DATA%TYPE,
```

```
P_KOD IN INTEGER,
EDIN OUT INTEGER
)
IS
X INTEGER;
CURSOR C_K IS
SELECT EDINIC FROM TABLE_USER WHERE
(DATA=P_DATA)AND(KOD_MATERIAL=P_KOD)AND(ADD_OR_DEL=0);
BEGIN
X:=0;
FOR V IN C_K LOOP
X:=X+V.EDINIC;
END LOOP;
EDIN:=X;
END Get_Edinic_DIL;
/
--////////////////////////////////////
create or replace procedure GET_NAME
(
P_NAME out VARCHAR2
)
IS
CURSOR C_K IS
SELECT TO_CHAR(user) from dual;
BEGIN
OPEN C_K;
FETCH C_K INTO P_NAME;
CLOSE C_K;
end GET_NAME;
/
--////////////////////////////////////
create or replace procedure CHASTO_PROD
(
P_KOD in INTEGER,
C_P out INTEGER
)
IS
CURSOR C_K IS
SELECT COUNT(*) FROM TABLE_USER WHERE (KOD_MATERIAL=P_KOD)AND(ADD_OR_DEL=0);
BEGIN
OPEN C_K;
FETCH C_K INTO C_P;
CLOSE C_K;
end CHASTO_PROD;
/
--////////////////////////////////////
create or replace procedure CHASTO_POK
(
P_KOD in INTEGER,
C_P out INTEGER
)
IS
CURSOR C_K IS
SELECT COUNT(*) FROM TABLE_USER WHERE (KOD_MATERIAL=P_KOD)AND(ADD_OR_DEL=1);
BEGIN
OPEN C_K;
FETCH C_K INTO C_P;
CLOSE C_K;
end CHASTO_POK;
/
```


Реализация алгоритма на DELPHI

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, DB, ADOdb, ComCtrls, Grids, DBGrids, RpCon, RpConDS,
  RpDefine, RpRave, ExtCtrls, frxClass, frxDBSet, ComObj;
type
  TForm1 = class(TForm)
    GroupBox1: TGroupBox;
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    Button1: TButton;
    TabSheet4: TTabSheet;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Button2: TButton;
    TabSheet5: TTabSheet;
    GroupBox2: TGroupBox;
    Label6: TLabel;
    GroupBox3: TGroupBox;
    Label13: TLabel;
    Edit7: TEdit;
    Button6: TButton;
    GroupBox7: TGroupBox;
    Label14: TLabel;
    Label15: TLabel;
    Edit9: TEdit;
    Edit10: TEdit;
    Button7: TButton;
    GroupBox8: TGroupBox;
    Label16: TLabel;
    Edit11: TEdit;
    Button8: TButton;
    Label23: TLabel;
    Edit12: TEdit;
    TabSheet3: TTabSheet;
    PageControl2: TPageControl;
    TabSheet6: TTabSheet;
    TabSheet7: TTabSheet;
    GroupBox4: TGroupBox;
    Label7: TLabel;
    Label8: TLabel;
    Edit4: TEdit;
    Edit5: TEdit;
    Button3: TButton;
    GroupBox6: TGroupBox;
    Label19: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Edit6: TEdit;
    Button4: TButton;
    Edit8: TEdit;
    ComboBox2: TComboBox;
    StatusBar1: TStatusBar;
    GroupBox9: TGroupBox;
    Label27: TLabel;
    ComboBox4: TComboBox;
    Label30: TLabel;
    ComboBox5: TComboBox;
    Label32: TLabel;
    Label33: TLabel;
    Edit16: TEdit;
    Label34: TLabel;
    Edit17: TEdit;
    GroupBox13: TGroupBox;
    PageControl3: TPageControl;
    TabSheet8: TTabSheet;
    DBGrid1: TDBGrid;
    Label24: TLabel;
    GroupBox11: TGroupBox;
    Label28: TLabel;
    Label29: TLabel;
    Edit13: TEdit;
    Edit14: TEdit;
    GroupBox12: TGroupBox;
    Label31: TLabel;
    Edit15: TEdit;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    Button13: TButton;
    ADOConnection1: TADOConnection;
    ADOStoredProc1: TADOStoredProc;
    ADOQuery1: TADOQuery;
    ADOStoredProc2: TADOStoredProc;
    ADOStoredProc3: TADOStoredProc;
    ADOStoredProc4: TADOStoredProc;
    DataSource1: TDataSource;
    ADOTable1: TADOTable;
    TabSheet9: TTabSheet;
    DBGrid2: TDBGrid;
    DBGrid3: TDBGrid;
```

```
DBGrid4: TDBGrid;
Label12: TLabel;
Label25: TLabel;
Label35: TLabel;
DataSource2: TDataSource;
ADOTable2: TADOTable;
DataSource3: TDataSource;
ADOTable3: TADOTable;
DataSource4: TDataSource;
ADOTable4: TADOTable;
ADOStoredProc5: TADOStoredProc;
GroupBox10: TGroupBox;
RadioButton3: TRadioButton;
Label36: TLabel;
Edit19: TEdit;
GroupBox14: TGroupBox;
Label17: TLabel;
ListBox1: TListBox;
Label20: TLabel;
Button9: TButton;
Label18: TLabel;
ListBox2: TListBox;
Label19: TLabel;
ListBox3: TListBox;
Label21: TLabel;
Label22: TLabel;
Button12: TButton;
Memo1: TMemo;
RadioButton4: TRadioButton;
Label5: TLabel;
Edit20: TEdit;
Panel1: TPanel;
Button14: TButton;
Label26: TLabel;
RadioButton5: TRadioButton;
Edit21: TEdit;
Label37: TLabel;
ADOStoredProc6: TADOStoredProc;
ADOStoredProc7: TADOStoredProc;
Button10: TButton;
frxDBDataset1: TfrxDBDataset;
Button5: TButton;
Button11: TButton;
frxReport1: TfrxReport;
ADOStoredProc8: TADOStoredProc;
ADOStoredProc9: TADOStoredProc;
procedure Button9Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button6Click(Sender: TObject);
procedure Button7Click(Sender: TObject);
procedure Button8Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Button11Click(Sender: TObject);
procedure VISUAL(S: AnsiString; LB: TListBox; ADOQ: TADOQuery;
ADOQ: TADOConnection; S1: AnsiString; L: TLabel);
procedure Button13Click(Sender: TObject);
procedure ComboBox4Enter(Sender: TObject);
procedure COMBOS(Q: TADOQuery; ADOC: TADOConnection; S,S1,S2: AnsiString;
CB: TComboBox; CB2: TComboBox);
procedure USERADD(M: AnsiString; S: AnsiString; EDID: integer; DAT: Tdatetime;
PROC: TADOStoredProc; ADOC: TADOConnection);
procedure SHOWVALUES(S3: AnsiString; Q: TADOQuery; ADOC: TADOConnection);
procedure SHOWVALUES2(S3: AnsiString; Q: TADOQuery; ADOC: TADOConnection);
procedure ComboBox5Enter(Sender: TObject);
procedure Button14Click(Sender: TObject);
procedure Edit19Enter(Sender: TObject);
procedure Edit20Enter(Sender: TObject);
procedure Edit21Enter(Sender: TObject);
procedure Button12Click(Sender: TObject);
procedure Button10Click(Sender: TObject);
procedure ComboBox4Change(Sender: TObject);
procedure ComboBox5Change(Sender: TObject);
procedure ComboBox2Enter(Sender: TObject);
end;
var
Form1: TForm1;
CEN: real;
KEYS,ZS: integer;

implementation

{$R *.dfm}
////////////////////////////////////
////////////////////////////////////+Просмотреть статистику ВЛ+////////////////////////////////////
////////////////////////////////////
procedure TForm1.VISUAL(S: AnsiString; LB: TListBox; ADOQ: TADOQuery;
ADOQ: TADOConnection; S1: AnsiString; L: TLabel);
var i: longint;
flag: boolean;
buf: integer;
begin
LB.Clear;
ADOQ.Active:=false;
ADOQ.Connection:=ADOC;
ADOQ.SQL.Clear;
ADOQ.SQL.Add(S);
```

```
ADOQ.Active:=true;
//////////Сам вывод////////////////////////////////////////
buf:=0;
ADOQ.First;
while (ADOQ.Eof=false) do
begin
    flag:=false;
    for i:=0 to LB.Items.Count-1 do
        if LB.Items[i]=ADOQ.FieldByName(S1).AsString
            then flag:=true;
    if flag=false then
        LB.Items.Add(ADOQ.FieldByName(S1).AsString);
        ADOQ.Next;
        buf:=buf+1;
    end;
    L.Caption:='Bcero:'+VarToStr(buf);
end;
//////////
procedure TForm1.Button9Click(Sender: TObject);
begin
try
    Form1.VISUAL('select material from SYSTEM.table_material',
        ListBox1,Form1.ADOQuery1,Form1.ADOConnection1,'material',Label20);
    Form1.VISUAL('select sklad from system.table_sklad',
        ListBox2,Form1.ADOQuery1,Form1.ADOConnection1,'sklad',Label21);
    Form1.VISUAL('select fio from system.table_fio',
        ListBox3,Form1.ADOQuery1,Form1.ADOConnection1,'fio',Label22);
except
    on E: EOLEException do ShowMessage('Не удалось просмотреть статистику: '+E.Message)
    else begin end;
end;
end;
//////////
//////////+Разъединение с БД////////////////////////////////////////
procedure TForm1.Button2Click(Sender: TObject);
begin
    ADOConnection1.Connected:=false;
    Form1.Button1.Enabled:=true;
    Form1.Button2.Enabled:=false;
    Form1.Button3.Enabled:=false;
    Form1.Button4.Enabled:=false;
    Form1.Button5.Enabled:=false;
    Form1.Button6.Enabled:=false;
    Form1.Button7.Enabled:=false;
    Form1.Button8.Enabled:=false;
    Form1.Button9.Enabled:=false;
    Form1.Button10.Enabled:=false;
    Form1.Button11.Enabled:=false;
    Form1.Button12.Enabled:=false;
    Form1.Button13.Enabled:=false;
    Form1.Button14.Enabled:=false;
    Form1.ComboBox4.Enabled:=false;
    Form1.ComboBox2.Enabled:=false;
    Form1.ComboBox5.Enabled:=false;
end;
//////////
//////////+Инициализация////////////////////////////////////////
procedure TForm1.FormCreate(Sender: TObject);
begin
    Form1.Left:=160;
    Form1.Top:=117;
    Form1.Button2.Enabled:=false;
    Form1.Button3.Enabled:=false;
    Form1.Button4.Enabled:=false;
    Form1.Button5.Enabled:=false;
    Form1.Button6.Enabled:=false;
    Form1.Button7.Enabled:=false;
    Form1.Button8.Enabled:=false;
    Form1.Button9.Enabled:=false;
    Form1.Button10.Enabled:=false;
    Form1.Button11.Enabled:=false;
    Form1.Button12.Enabled:=false;
    Form1.Button13.Enabled:=false;
    Form1.Button14.Enabled:=false;
    Form1.ComboBox4.Enabled:=false;
    Form1.ComboBox2.Enabled:=false;
    Form1.ComboBox5.Enabled:=false;
    Form1.RadioButton1.Checked:=true;
    Form1.RadioButton3.Checked:=true;
end;
//////////
//////////+Соединение с БД////////////////////////////////////////
procedure TForm1.Button1Click(Sender: TObject);
begin
try
    ADOConnection1.ConnectionString:='Provider=MSDAORA.1;';
    ADOConnection1.ConnectionString:=ADOConnection1.ConnectionString+
        'Persist Security Info=false;';
    ADOConnection1.ConnectionString:=ADOConnection1.ConnectionString+
        'User ID='+Edit2.Text+'';
    ADOConnection1.ConnectionString:=ADOConnection1.ConnectionString+
        'Password='+Edit3.Text+'';
    ADOConnection1.ConnectionString:=ADOConnection1.ConnectionString+
        'Data Source='+Edit1.Text+'';
    ADOConnection1.LoginPrompt:=false;
    ADOConnection1.Provider:='MSDAORA.1';
    ADOConnection1.Connected:=true;
```

```
if ADOConnection1.Connected=true then
begin
  ShowMessage('Соединение установлено!');
  Form1.Button1.Enabled:=false;
  Form1.Button2.Enabled:=true;
  Form1.Button3.Enabled:=true;
  Form1.Button4.Enabled:=true;
  Form1.Button5.Enabled:=true;
  Form1.Button6.Enabled:=true;
  Form1.Button7.Enabled:=true;
  Form1.Button8.Enabled:=true;
  Form1.Button9.Enabled:=true;
  Form1.Button10.Enabled:=true;
  Form1.Button11.Enabled:=true;
  Form1.Button12.Enabled:=true;
  Form1.Button13.Enabled:=true;
  Form1.Button14.Enabled:=true;
  Form1.ComboBox4.Enabled:=true;
  Form1.ComboBox2.Enabled:=true;
  Form1.ComboBox5.Enabled:=true;
  ///////////////////////////////////Выдаем сообщение пользователю - КТО ОН////////////////////////////////////
  Form1.ADOStoredProc5.Connection:=Form1.ADOConnection1;
  Form1.ADOStoredProc5.ExecProc;
  if Form1.ADOStoredProc5.Parameters[0].Value='SYSTEM' then
  begin
    Form1.StatusBar1.Panels[1].Text:='Администратор: '+
      Form1.ADOStoredProc5.Parameters[0].Value;
    ADOTable2.Active:=false;
    ADOTable2.TableName:='SYSTEM.TABLE_FIO';
    ADOTable2.Connection:=Form1.ADOConnection1;
    ADOTable3.Active:=false;
    ADOTable3.TableName:='SYSTEM.TABLE_SKLAD';
    ADOTable3.Connection:=Form1.ADOConnection1;
    ADOTable4.Active:=false;
    ADOTable4.TableName:='SYSTEM.TABLE_MATERIAL';
    ADOTable4.Connection:=Form1.ADOConnection1;
    Form1.ADOTable2.Active:=true;
    Form1.ADOTable3.Active:=true;
    Form1.ADOTable4.Active:=true;
  end
else
begin
  Form1.StatusBar1.Panels[1].Text:='Пользователь: '+
    Form1.ADOStoredProc5.Parameters[0].Value;
  end;
  ///////////////////////////////////
  ADOTable1.Active:=false;
  ADOTable1.TableName:='SYSTEM.TABLE_USER';
  ADOTable1.Connection:=Form1.ADOConnection1;
  Form1.ADOTable1.Active:=true;
end
else
  ShowMessage('Невозможно установить соединение!')
except
on E: EOLEException do ShowMessage('Не удалось соединиться с сервером: ' + E.Message)
else begin end;
end;
end;
////////////////////////////////////
procedure TForm1.Edit19Enter(Sender: TObject);
begin
  Form1.RadioButton3.Checked:=true;
end;
////////////////////////////////////
procedure TForm1.Edit20Enter(Sender: TObject);
begin
  Form1.RadioButton4.Checked:=true;
end;
////////////////////////////////////
procedure TForm1.Edit21Enter(Sender: TObject);
begin
  Form1.RadioButton5.Checked:=true;
end;
////////////////////////////////////
////////////////////////////////////+Состояние дел на складах////////////////////////////////////
////////////////////////////////////
procedure TForm1.Button14Click(Sender: TObject);
var COUNT: integer;
    ID: integer;
begin
try
  if Form1.RadioButton5.Checked=true then
  begin
    ///////////////////////////////////Приход расход////////////////////////////////////
    Memo1.Clear;
    Memo1.Lines.Add('Приход товара:');
    ///////////////////////////////////Запрос на таблицу материалов////////////////////////////////////
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.Connection:=Form1.ADOConnection1;
    Form1.ADOQuery1.Parameters.Clear;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('SELECT * FROM SYSTEM.TABLE_MATERIAL');
    Form1.ADOQuery1.Active:=true;
    Form1.ADOQuery1.First;
    while (Form1.ADOQuery1.Eof=false) do
    begin
      ID:=Form1.ADOQuery1.FieldByName('ID').Value;
      Memo1.Lines.Add('Код материала: '+VarToStr(ID));
      Form1.ADOStoredProc6.Connection:=Form1.ADOConnection1;
      Form1.ADOStoredProc6.Parameters[0].Value:=StrToDateTime(Edit21.Text);
```

```
Form1.ADOStoredProc6.Parameters[1].Value:=ID;
Form1.ADOStoredProc6.ExecProc;
Form1.ADOStoredProc8.Connection:=Form1.ADOConnection1;
Form1.ADOStoredProc8.Parameters[0].Value:=StrToDateTime(Edit21.Text);
Form1.ADOStoredProc8.Parameters[1].Value:=ID;
Form1.ADOStoredProc8.ExecProc;
Mem1.Lines.Add('Прибыло всего единиц: '+VarToStr(Form1.ADOStoredProc6.Parameters[2].Value));
Mem1.Lines.Add('Убыло всего единиц: '+VarToStr(Form1.ADOStoredProc8.Parameters[2].Value));
Form1.ADOQuery1.Next;
end;
end;
if Form1.RadioButton3.Checked=true then
begin
//////////Частота продаваемости//////////
Mem1.Clear;
Mem1.Lines.Add('Частота покупаемости материала > раз: '+Edit19.Text);
//////////Запрос на таблицу материалов//////////
Form1.ADOQuery1.Active:=false;
Form1.ADOQuery1.Connection:=Form1.ADOConnection1;
Form1.ADOQuery1.Parameters.Clear;
Form1.ADOQuery1.SQL.Clear;
Form1.ADOQuery1.SQL.Add('SELECT * FROM SYSTEM.TABLE_MATERIAL');
Form1.ADOQuery1.Active:=true;
Form1.ADOQuery1.First;
while (Form1.ADOQuery1.Eof=false) do
begin
ID:=Form1.ADOQuery1.FieldByName('ID').Value;
Form1.ADOStoredProc7.Connection:=Form1.ADOConnection1;
Form1.ADOStoredProc7.Parameters[0].Value:=ID;
Form1.ADOStoredProc7.ExecProc;
if Form1.ADOStoredProc7.Parameters[1].Value>StrToInt(Edit19.Text) then
begin
Mem1.Lines.Add('Код материала: '+VarToStr(ID));
end;
Form1.ADOQuery1.Next;
end;
end;
end;
if Form1.RadioButton4.Checked=true then
begin
//////////Частота покупаемости//////////
Mem1.Clear;
Mem1.Lines.Add('Частота продаваемости материала > раз: '+Edit20.Text);
//////////Запрос на таблицу материалов//////////
Form1.ADOQuery1.Active:=false;
Form1.ADOQuery1.Connection:=Form1.ADOConnection1;
Form1.ADOQuery1.Parameters.Clear;
Form1.ADOQuery1.SQL.Clear;
Form1.ADOQuery1.SQL.Add('SELECT * FROM SYSTEM.TABLE_MATERIAL');
Form1.ADOQuery1.Active:=true;
Form1.ADOQuery1.First;
while (Form1.ADOQuery1.Eof=false) do
begin
ID:=Form1.ADOQuery1.FieldByName('ID').Value;
Form1.ADOStoredProc9.Connection:=Form1.ADOConnection1;
Form1.ADOStoredProc9.Parameters[0].Value:=ID;
Form1.ADOStoredProc9.ExecProc;
if Form1.ADOStoredProc9.Parameters[1].Value>StrToInt(Edit20.Text) then
begin
Mem1.Lines.Add('Код материала: '+VarToStr(ID));
end;
Form1.ADOQuery1.Next;
end;
end;
end;
except
on E: EOLEException do ShowMessage('Не удалось сгенерировать отчет: '+E.Message)
else ShowMessage('Введите данные правильно!');
end;
end;
//////////+Отчет пользователя+//////////
procedure TForm1.Button12Click(Sender: TObject);
begin
try
Form1.frxDBDataset1.DataSet:=ADOTable1;
Form1.frxReport1.LoadFromFile('USER.fr3');
Form1.frxReport1.ShowReport(true);
except
on E: EOLEException do ShowMessage('Не удалось создать отчет: '+E.Message)
else ShowMessage('Не могу открыть файл USER.fr3!');
end;
end;
//////////+Отчет админа по 3С+//////////
procedure TForm1.Button10Click(Sender: TObject);
begin
try
Form1.frxDBDataset1.DataSet:=ADOTable2;
Form1.frxReport1.LoadFromFile('ADM1.fr3');
Form1.frxReport1.ShowReport(true);
except
on E: EOLEException do ShowMessage('Не удалось создать отчет: '+E.Message)
else ShowMessage('Не могу открыть файл ADM1.fr3!');
end;
end;
//////////+Отчет администратора по складам+//////////
procedure TForm1.Button5Click(Sender: TObject);
```

```
begin
try
Form1.frxDBDataset1.DataSet:=ADOTable3;
Form1.frxReport1.LoadFromFile('ADM2.fr3');
Form1.frxReport1.ShowReport(true);
except
on E: EOLEException do ShowMessage('Не удалось создать отчет: '+E.Message)
else ShowMessage('Не могу открыть файл ADM2.fr3!');
end;
end;
//+Отчет администратора по товарам+//
//+Отчет администратора по товарам+//
//+Отчет администратора по товарам+//
procedure TForm1.Button11Click(Sender: TObject);
begin
try
Form1.frxDBDataset1.DataSet:=ADOTable4;
Form1.frxReport1.LoadFromFile('ADM3.fr3');
Form1.frxReport1.ShowReport(true);
except
on E: EOLEException do ShowMessage('Не удалось создать отчет: '+E.Message)
else ShowMessage('Не могу открыть файл ADM3.fr3!');
end;
end;
//+Вызов хранимой процедуры+//
//+Вызов хранимой процедуры+//
//+Вызов хранимой процедуры+//
procedure TForm1.USERADD(M: AnsiString; S: AnsiString; EDID: integer; DAT: Tdatetime;
PROC: TADOStoredProc; ADOC: TADOConnection);
begin
PROC.Connection:=ADOC;
PROC.Parameters[0].Value:=M;
PROC.Parameters[1].Value:=S;
PROC.Parameters[2].Value:=EDID;
PROC.Parameters[3].Value:=DAT;
PROC.ExecProc;
end;
//+Пользователь добавляет запись о товаре+//
//+Пользователь добавляет запись о товаре+//
//+Пользователь добавляет запись о товаре+//
procedure TForm1.Button13Click(Sender: TObject);
begin
try
if Form1.RadioButton1.Checked=true then
begin
//Добавляем//
USERADD(ComboBox4.Text,ComboBox5.Text,StrToInt(Edit16.Text),StrToDateTime(Edit17.Text),
Form1.ADOStoredProc1,Form1.ADOConnection1);
end
else
begin
//Удаляем//
USERADD(ComboBox4.Text,ComboBox5.Text,StrToInt(Edit16.Text),StrToDateTime(Edit17.Text),
Form1.ADOStoredProc2,Form1.ADOConnection1);
end;
ADOTable1.Active:=false;
ADOTable1.Active:=true;
except
on E: EOLEException do ShowMessage('Не удалось добавить запись: '+E.Message)
else ShowMessage('Введите данные правильно!');
end;
end;
//+Показываем дополнительную информацию+//
//+Показываем дополнительную информацию+//
//+Показываем дополнительную информацию+//
procedure TForm1.SHOWVALUES(S3: AnsiString;Q: TADOQuery; ADOC: TADOConnection);
var I: real;
begin
Q.Active:=false;
Q.Connection:=ADOC;
Q.SQL.Clear;
Q.SQL.Add('SELECT CENA,KEYS FROM SYSTEM.TABLE_MATERIAL WHERE MATERIAL=:S3');
Q.Parameters.Clear;
Q.Parameters.Add;
Q.Parameters[0].DataType:=ftString;
Q.Parameters[0].Direction:=pdInput;
Q.Parameters[0].Name:='S3';
Q.Parameters[0].Value:=S3;
Q.Active:=true;
Q.First;
CEN:=Q.Fields[0].AsFloat;
KEYS:=Q.Fields[1].AsInteger;
end;
//+Показываем дополнительную информацию+//
//+Показываем дополнительную информацию+//
//+Показываем дополнительную информацию+//
procedure TForm1.SHOWVALUES2(S3: AnsiString; Q: TADOQuery; ADOC: TADOConnection);
begin
Q.Active:=false;
Q.Connection:=ADOC;
Q.SQL.Clear;
Q.SQL.Add('SELECT KEYS FROM SYSTEM.TABLE_SKLAD WHERE SKLAD=:S3');
Q.Parameters.Clear;
Q.Parameters.Add;
Q.Parameters[0].DataType:=ftString;
Q.Parameters[0].Direction:=pdInput;
Q.Parameters[0].Name:='S3';
Q.Parameters[0].Value:=S3;
Q.Active:=true;
Q.First;
ZS:=Q.Fields[0].AsInteger;
end;
```

```
////////////////////////////////////
////////////////////////////////////+Вывод в Combobox+////////////////////////////////////
////////////////////////////////////
procedure TForm1.COMBOS(Q: TADOQuery; ADOC: TADOConnection; S,S1,S2: AnsiString;
                        CB: TComboBox; CB2: TComboBox);
begin
  Q.Active:=false;
  Q.Connection:=ADOC;
  Q.SQL.Clear;
  Q.SQL.Add(S);
  //////////////////////////////////////
  Q.Parameters.Clear;
  if S2<>'' then //Второй раз вызывается для склада - там параметр не нужен
  begin
    Q.Parameters.Add;
    Q.Parameters[0].Name:=S2;
    Q.Parameters[0].DataType:=ftFixedChar;
    Q.Parameters[0].Direction:=pdInput;
    Q.Parameters[0].Size:=255;
    Q.Parameters[0].Value:=CB2.Text;
  end;
  Q.Active:=true;
  Q.First;
  CB.Clear;
  while (Q.Eof=false) do
  begin
    CB.Items.Add(Q.FieldByName(S1).Value);
    Q.Next;
  end;
end;
////////////////////////////////////
procedure TForm1.ComboBox4Enter(Sender: TObject);
begin
  Form1.COMBOS(ADOQuery1,ADOConnection1,
               'SELECT MATERIAL FROM SYSTEM.TABLE_MATERIAL WHERE TABLE_MATERIAL.KEYS=(SELECT TABLE_SKLAD.ID FROM SYSTEM.TABLE_SKLAD WHERE
SKLAD=:SKL)', 'MATERIAL', 'SKL', ComboBox4, ComboBox5);
end;
////////////////////////////////////
procedure TForm1.ComboBox5Enter(Sender: TObject);
begin
  ComboBox4.Clear;
  Edit13.Text:='';
  Edit14.Text:='';
  Form1.COMBOS(ADOQuery1,ADOConnection1,
               'SELECT SKLAD FROM SYSTEM.TABLE_SKLAD', 'SKLAD', '', ComboBox5, ComboBox5);
end;
////////////////////////////////////
procedure TForm1.ComboBox4Change(Sender: TObject);
var S: AnsiString;
begin
  S:=Form1.ComboBox4.Text;
  if ComboBox4.Text<>'' then
    Form1.SHOWVALUES(ComboBox4.Text,ADOQuery1,ADOConnection1);
  Edit13.Text:=VarToStr(CEN);
  Edit14.Text:=VarToStr(KEYS);
end;
////////////////////////////////////
procedure TForm1.ComboBox5Change(Sender: TObject);
var S: AnsiString;
begin
  S:=Form1.ComboBox5.Text;
  if ComboBox5.Text<>'' then
    Form1.SHOWVALUES(ComboBox5.Text,ADOQuery1,ADOConnection1);
  Edit15.Text:=VarToStr(ZS);
end;
////////////////////////////////////
////////////////////////////////////+Добавление записи о ЗС+////////////////////////////////////
////////////////////////////////////
procedure TForm1.Button3Click(Sender: TObject);
begin
  try
    Form1.ADOStoredProc3.Active:=false;
    Form1.ADOStoredProc3.Connection:=Form1.ADOConnection1;
    Form1.ADOStoredProc3.Parameters.ParamByName('P_FIO').Value:=Edit5.Text;
    Form1.ADOStoredProc3.Parameters.ParamByName('P_SKLAD').Value:=Edit4.Text;
    Form1.ADOStoredProc3.ExecProc;
    ADOTable2.Active:=false;
    ADOTable3.Active:=false;
    ADOTable2.Active:=true;
    ADOTable3.Active:=true;
    ShowMessage('Запрос завершился правильно!');
  except
    on E: EOLEException do ShowMessage('Не удалось добавить запись: '+E.Message)
    else ShowMessage('Введите данные правильно!');
  end;
end;
////////////////////////////////////
////////////////////////////////////+Добавление записи о материале+////////////////////////////////////
////////////////////////////////////
procedure TForm1.Button4Click(Sender: TObject);
begin
  try
    Form1.ADOStoredProc4.Active:=false;
    Form1.ADOStoredProc4.Connection:=Form1.ADOConnection1;
    Form1.ADOStoredProc4.Parameters[0].Value:=Edit6.Text;
    Form1.ADOStoredProc4.Parameters[1].Value:=Form1.ComboBox2.Text;
    Form1.ADOStoredProc4.Parameters[2].Value:=StrToFloat(Edit8.Text);
    Form1.ADOStoredProc4.ExecProc;
    ADOTable4.Active:=false;
```

```
ADOTable4.Active:=true;
ShowMessage('Вы добавили новое наименование товара!');
except
  on E: EOLEException do ShowMessage('Не удалось добавить запись: '+E.Message)
  else ShowMessage('Введите данные правильно!');
end;
end;
////////////////////////////////////
////////////////////////////////////+Меняем пароль администратору+////////////////////////////////////
////////////////////////////////////
procedure TForm1.Button6Click(Sender: TObject);
begin
  if Edit7.Text=Edit12.Text then
  begin
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.Connection:=Form1.ADOConnection1;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('ALTER USER SYSTEM IDENTIFIED BY '+Edit7.Text);
    try
      Form1.ADOQuery1.ExecSQL;
      ShowMessage('Пароль для администратора SYSTEM сменил!');
    except
      ShowMessage('Не удалось сменить пароль!');
    end;
  end
  else
  begin
    ShowMessage('Повторите пароль - два значения не совпадают!');
    Edit12.Text:='';
    Edit7.Text:='';
  end;
end;
////////////////////////////////////
////////////////////////////////////+Удаляем пользователя+////////////////////////////////////
////////////////////////////////////
procedure TForm1.Button8Click(Sender: TObject);
begin
  Form1.ADOQuery1.Active:=false;
  Form1.ADOQuery1.Connection:=Form1.ADOConnection1;
  Form1.ADOQuery1.SQL.Clear;
  Form1.ADOQuery1.SQL.Add('DROP USER '+Edit11.Text);
  try
    Form1.ADOQuery1.ExecSQL;
    ShowMessage('Пользователь удален!');
  except
    ShowMessage('Не удалось удалить пользователя - возможно не хватает полномочий!');
  end;
end;
////////////////////////////////////
////////////////////////////////////+Создаем нового пользователя+////////////////////////////////////
////////////////////////////////////
procedure TForm1.Button7Click(Sender: TObject);
begin
  Form1.ADOQuery1.Active:=false;
  Form1.ADOQuery1.Connection:=Form1.ADOConnection1;
  try
    //////////////////////////////////Создаем пользователя////////////////////////////////////
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('CREATE USER '+Edit9.Text+' IDENTIFIED BY '+
    Edit10.Text+' DEFAULT TABLESPACE USERS');
    Form1.ADOQuery1.ExecSQL;
    //////////////////////////////////Даем ему привелегии////////////////////////////////////
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT CREATE SESSION TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    //Просмотр таблиц
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT SELECT ON SYSTEM.TABLE_USER TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT INSERT ON SYSTEM.TABLE_USER TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT SELECT ON SYSTEM.TABLE_MATERIAL TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT SELECT ON SYSTEM.TABLE_SKLAD TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT EXECUTE ON SYSTEM.GET_NAME TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT EXECUTE ON SYSTEM.Proc_User_Add TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.Active:=false;
    Form1.ADOQuery1.SQL.Clear;
    Form1.ADOQuery1.SQL.Add('GRANT EXECUTE ON SYSTEM.Proc_User_Del TO '+Edit9.Text);
    Form1.ADOQuery1.ExecSQL;
    Form1.ADOQuery1.ExecSQL;
    ShowMessage('Пользователь '+Edit9.Text+' создан!');
```



```
except
  on E: EOLEException do ShowMessage('Не удалось создать пользователя: '+E.Message)
  else begin end;
end;
end;
////////////////////////////////////
procedure TForm1.ComboBox2Enter(Sender: TObject);
begin
  try
    Form1.COMBOS(ADOQuery1,ADOConnection1,
      'SELECT SKLAD FROM SYSTEM.TABLE_SKLAD', 'SKLAD', '', ComboBox2, ComboBox2);
  except
    on E: EOLEException do ShowMessage('Не удалось обновить: '+E.Message)
    else ShowMessage('Введите данные правильно!');
  end;
end;
////////////////////////////////////END////////////////////////////////////
end.
```