

Kierunek: **ETI**

Przedmiot: **Programowanie w środowisku RAD - Delphi**

Rok **III** Semestr 5

Ćwiczenie 7 - Aplikacja do przeglądania pojedynczej tabeli.

1. Przygotuj (załóż) nowe katalogi dla tworzonej bazy danych i opracowywanej aplikacji.

Utwórz plik bazy danych w formacie INTERBASE o nazwie: **SKOKI.GDB**. Wygeneruj przy pomocy programu IBConsole bazę zawierającą dwie tabele DRUZYNY i ZAWODNICY. Zastosuj instrukcje SQL:

```
CREATE TABLE Druzyzny
(
  ID_DRU   VARCHAR(4)  NOT NULL PRIMARY KEY,
  NAZWA    VARCHAR(20) NOT NULL,
  MIASTO   VARCHAR(20)
);
CREATE TABLE Zawodnicy
(
  ID_ZAW      VARCHAR(4)  NOT NULL PRIMARY KEY,
  NAZWISKO    VARCHAR(20) NOT NULL,
  IMIE        VARCHAR(20),
  WAGA        NUMERIC(3, 0),
  WZROST      NUMERIC(3, 0),
  DATA_UR    TIMESTAMP,
  ID_DRUZYNY  VARCHAR(4)  REFERENCES Druzyzny
);
```

2. Opracuj prosty program obsługi bazy.

Do budowy aplikacji należy zastosować komponenty:

z palety Interbase: IBDatabase, IBTransaction i dwa IBTable

z palety Data Access: 2 komponenty DataSource

z palety Data Controls: 2 komponenty DBGrid

Komponenty **DBGrid** należy rozmieścić jeden pod drugim. Pozostałe komponenty są niewizualne i mogą być rozmieszczone na formacie projektowej dowolnie.

Dla opisu tabel dodać dwie etykiety **Label** i rozmieścić nad komponentami **DBGrid**.

Skonfigurowanie właściwości komponentów powinno być następujące:

IBDatabase1:

 DatabaseName: nazwa i ścieżka pliku bazy

 DefaultTransaction: wybrać IBTransaction1

 Connected: True

IBTable1:

 DataBase: wybrać IBDatabase1

 TableName: wybrać DRUZYNY

 Active: True

IBTable2:

 DataBase: wybrać IBDatabase1

 TableName: wybrać ZAWODNICY

 Active: True

DataSource1:

DataSet: wybrać IBTable1

DataSource2:

DataSet: wybrać IBTable2

DBGrid1:

DataSource: wybrać DataSource1

DBGrid2:

DataSource: wybrać DataSource2

Label1:

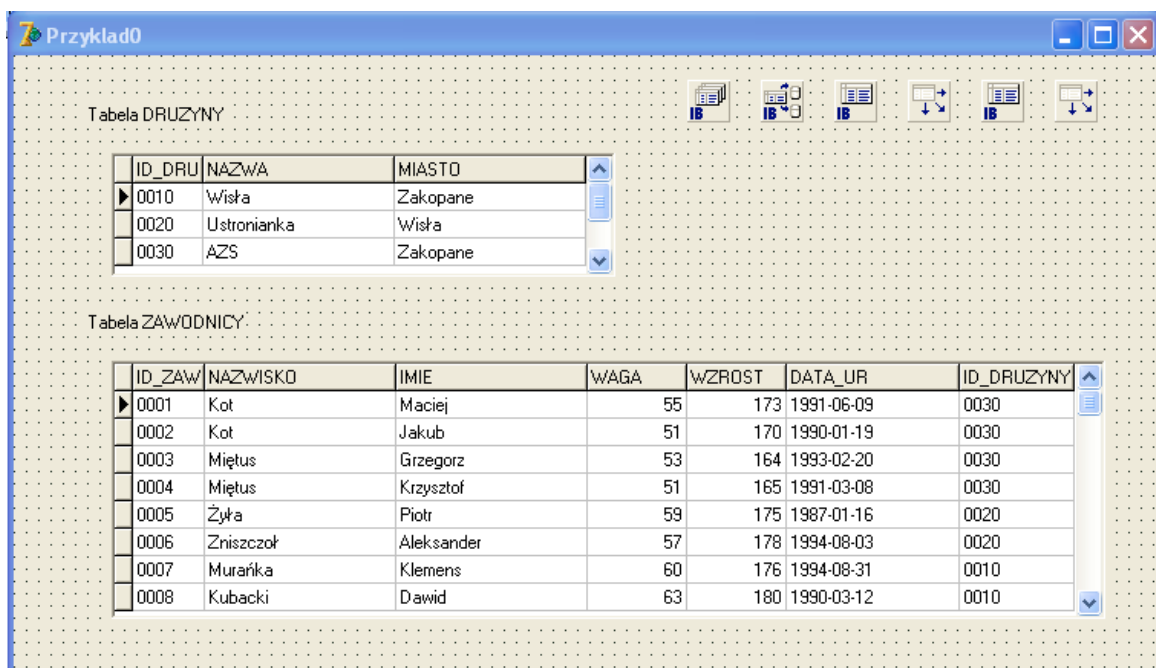
Caption: 'Tabela DRUZYNY'

Label2:

Caption: 'Tabela ZAWODNICY'

Wpisywanie wartości właściwości **Connected** lub **Active** może powodować wyświetlanie okienka logowania do bazy danych. Należy wpisywać nazwę i hasło użytkownika uprawnionego do korzystania z bazy. Standardowo nazwa użytkownika *SYSDBA* oraz hasło *masterkey*.

Wykorzystaj zbudowaną aplikację do wpisania danych do tabel bazy. Postać okienka aplikacji (w fazie projektowania) powinna po wpisaniu danych być podobna do przedstawionej poniżej:



UWAGA. Aplikacja nie zawiera żadnej obsługi błędów. Przy wpisywaniu danych należy unikać podawania danych niezgodnych z definicją bazy. Np. należy wpisać najpierw dane do tabeli *Druzyny* a dopiero potem dane zawodników.

3. Uzupełnij program o możliwość wyznaczenia (inicjowanego naciśnięciem przycisku) średniej wagi osób urodzonych przed 1991r.

Dołącz potrzebne komponenty **Button1** i **Label3**.

Dopisz metodę obsługi zdarzenia OnClick przycisku. Proponowany zapis:

```
procedure TForm1.Button1Click(Sender: TObject);
var suma, liczba:integer;
    dzien, miesiac, rok: word;
```

```

begin
IBTable2.First;
suma:=0;
liczba:=0;

while not IBTable2.EOF do
begin
DecodeDate(IBTable2.FieldByName('DATA_UR').AsDateTime,
            rok, miesiac, dzien);
if rok < 1980 then
begin
suma:=suma+IBTable2.Fields[3].AsInteger;
liczba:=liczba+1;
end;
IBTable2.Next;
end;
Label3.caption:=FloatToStr(suma/liczba);
end;

```

Zauważ sposób odwoływania się do pól tabeli przy pomocy metody **FieldByName** lub tablicy **Fields**.

4. Zmodyfikuj aplikację. Dołącz nowy przycisk i wykorzystaj go do wyświetlania imienia, nazwiska oraz daty urodzenia osoby najlżejszej.
5. Zmodyfikuj aplikację. Dołącz nowy przycisk i wykorzystaj go do wyświetlania komunikatu zawierającego nazwę tabeli i nazwy jej pól.
Propozycja zapisu metody:

```

procedure TForm1.Button2Click(Sender: TObject);
var
i: Integer;
Info: String;
begin
Info := 'Pola tabeli ' + IBTable2.TableName + ' to:'#13#10#13#10;
for i := 0 to IBTable2.FieldCount - 1 do
Info := Info + IBTable2.Fields[i].FieldName + #13#10;
ShowMessage(Info);
end;

```

Uzupełnij wyświetlane wyniki o dane zapisane we właściwościach **FieldType**, **Size** i **DataSize** obiektowych elementów tablicy **Fields**.

6. Uzupełnij program o nową etykietę, a następnie o metodę obsługującą zdarzenie **OnStateChange** dla komponentu **DataSource** o zapisie:

```

procedure TForm1.DataSource2StateChange(Sender: TObject);
var Tytul:String;
begin
case IBTable2.State of
dsBrowse: Tytul:= 'Browse';
dsEdit: Tytul:= 'Edit';
dsInsert: Tytul:= 'Insert';
dsInactive: Tytul:= 'Inactive';
else
Tytul:= 'Inny stan';
end;
Label4.Caption := 'Stan tabeli: ' + Tytul;
end;

```

Zaobserwuj zmiany stanu tabeli w czasie jej obsługi.