Контрольная работа

В данном курсе все примеры приведены в ПО Delphi. При остуствии возможности использовать Delphi, Вы можете обратиться к СПО [Lazarus](https://ru.wikipedia.org/wiki/Lazarus" \t "blank).

Задание на контрольную работу:

1. Создать базу данных (БД), состоящую из 2-х заданных таблиц. Поля таблиц произвольные, но не менее четырех полей в каждой таблице, включая ключевое поле (поле типа +(Autoincrement)). В таблицу, которая при объединении будет подчиненной, необходимо включить поле, по которому эта таблица будет связана с первичным ключом главной таблицы (в рассматриваемом здесь примере это поле NFcl таблицы grp2).
2. Разработать Приложение для работы с БД, выполняющее те же функции, что и в приведенном ниже примере: просмотр, наполнение, редактирование таблиц; создание вычисляемого поля; организация связи главный-подчиненный двух таблиц; поиск данных по заданному критерию.

Поиск данных можете реализовать по любому полю, на Ваше усмотрение, подчиненной таблицы.

Варианты задания.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер варианта** | **Тема**  **Базы данных** | **Главная**  **таблица** | **Подчиненная**  **таблица** | **Что вычислять в**  **вычисляемом поле** |
| **0** | Группы факультета АЭС | Группы | Студенты | Средний балл за контрольные сроки (всего два контрольных срока) |
| **1** | Продовольственный магазин | Типы продуктов: молочные, мучные, соки, фрукты и т.д. | Продукты | Количество оставшихся на складе продуктов (поступило - продано) |
| **2** | Аптека | Тип медикаментов: антибиотики, болеутоляющие, жаропонижающие,… . | Медикаменты | Количество оставшихся на складе медикаментов (поступило - продано) |
| **3** | Прокат видеофиль-мов | Жанры: комедия, боевик, триллер, мелодрама, … . | Фильмы | Количество оставшихся дисков с данным фильмом (всего - отдано) |
| **4** | Театр | Тип спектакля:  опера, балет, концерт | Названия спектаклей | Количество проданных билетов (всего-оставшиеся) |
| **5** | Магазин спортивныхтоваров | Отделы: рыбалка, теннис, для горнолыжников,… | Товары | Количество проданного  товара  (поступило-осталось) |
| **6** | Кафе | Меню: закуски, первое блюдо, второе, десерт | Блюда | Цена со скидкой 15% |
| **7** | Отдел спиртных напитков | Тип напитка: вино, водка, пиво, … . | Напитки | Количество оставшегося напитка  (поступило- продано) |
| **8** | Книжный магазин | Отделы: худ. литература, учебники, детская литература, журналы… | Литература | Цена со скидкой 5% |
| **9** | Агенство недвижимости | Тип квартиры (по количеству комнат) | Квартиры | Общая площадь (жилая + нежилая) |

Требования к отчету:

* N варианта,
* структура таблиц,
* главная (Form1) форма приложения,
* Текст модуля.

Создание БД

Создание Приложения

1. Поместить на форму компоненты TTable, TDataSource, TDbGrid (следовательно, Table1, DataSource1, DbGrid1);
2. В инспекторе объектов устанавливаете следующие свойства:

Table1.DataBaseName:=’путь до базы’; {здесь указывается полный путь до папки с таблицами вашей базы данных}

Table1.TableName:=Fcl; {из выпадающего списка выбираете нужную таблицу. Если список пустой, скорее всего неправильно указан путь в предыдущем свойстве DataBaseName }

DataSource1.DataSet:=Table1;

DbGrid1.DataSource:=DataSource1;

Table1.Active:=True;

1. Поместить на форму TDBNavigator (DBNavigator1) из закладки Data Controls палитры компонентов. В инспекторе объектов устанавливаете следующие свойства:

DataSource:=DataSource1;

ShowHint:=True; (чтобы при подведении курсора мыши к какой-либо кнопке данного компонента отображалась подсказка)

1. Поместить на форму TLabel (Label1) из закладки Standard палитры компонентов. В инспекторе объектов устанавливаете следующие свойства:

Caption:=Факультет;

Font (шрифт)

После каждого этапа работы запускайте приложение на выполнение, чтобы посмотреть, что у Вас получилось и нет ли ошибок. (Запуск Приложения – Run или F9).

1. Создание вычисляемого поля у компонента TTable
   * В редакторе полей (открывается по двойному щелчку на компоненте):

New fields; - появится окно

* + В окне необходимо заполнить поля:

Name: Plata\_Obuch

Type: String

Выбрать: Calculated

* + В редакторе полей выделить новое поле Plata\_Obuch, в инспекторе объектов выбрать обработчик события OnGetText, двойной щелчок напротив, в появившейся заготовке процедуры написать код:

text:=IntToStr(Table1KolStd.asInteger-Table1Budget.asInteger);

1. Поместить на форму компоненты TQuery, TDataSource, TDbGrid (следовательно, Query1, DataSource2, DbGrid2);

В инспекторе объектов устанавливаете следующие свойства:

Query1.DataBaseName:=’путь до базы’;

DataSource2.DataSet:= Query1;

Query1.SQL: select \* from grp2; - это запрос к базе данных: выдать все (т.к. после

select - \*) поля из таблицы grp2. Такой запрос к БД

на выдачу данных называют SQL –запросом, т.к.

используется язык SQL. Про язык SQL и компонент TQuery

можно прочитать в файле Query.

Query1.Active:=True;

1. Поместить на форму компонент TMainMenu – из закладки Standard. С помощью редактора меню (двойной щелчок на компоненте) создать полосу главного меню (см. рис.). В пункте 1 два подпункта: Find (поиск) и Exit. Создайте обработчик Exit (двойной щелчок). Напишите код:

Close

Прежде, чем создавать обработчик пункта Find, добавьте к Приложению еще одну форму Form2: File \ New \ Form. Вернитесь к редактору меню и обработайте пункт Find. В обработчике напишите код: Form2.Show;

Запустите Приложение (на предложенный вопрос ответьте Yes) для проверки работы пунктов меню. У Вас теперь две формы в Приложении: Form1 – главная и Form2. Чтобы закрыть Приложение надо закрыть главную форму.

Теперь откройте Form2 (View \ Forms) и поместите компоненты: Edit1 и Button1. Создайте обработчик нажатия на кнопку и напишите следующий код:

Form1.Query1.Close;

Form1.Query1.SQL.Clear;

Form1.Query1.SQL.Add('select \* from grp2 where KolStd=:p');

Form1.Query1.ParamByName('p').AsString:=Edit1.Text;

// Про компонент TQuery и запросы с параметрами, как в данном случае, читайте в файле Query.

Form1.Query1.Open;

Form2.Close;

1. Поместить на форму компонент TRadioGroup– из закладки Standard.

В инспекторе объектов установите следующие свойства:

Caption:= Режимы

Items: Edit (В строке Items нажмите на кнопку, введите в появившемся окне

Insert значения, Ok)

Создайте обработчик OnClick компонента RadioGroup1, в котором напишите следующий код:

if Not(Query1.RequestLive) then begin

Query1.Active:=false;

Query1.RequestLive:=True;

Query1.Active:=true;

end;

if RadioGroup1.ItemIndex=0 then Query1.Edit;

if RadioGroup1.ItemIndex=1 then Query1.Insert;

Пояснения к коду:

* + Данные, выданные с помощью SQL-запроса по умолчанию являются данными только для чтения. Чтобы была возможность редактирования такой таблице, надо установить Query1.RequestLive=True; (по умолчанию Query1.RequestLive=False). Изменять значение этого свойства можно только при неактивном Query1 (т.е. при Query1.Active=false).
  + Свойство ItemIndex компонента TRadioGroup – номер выбранной радиокнопки данного компонента, нумерация начинается с 0. По умолчанию ItemIndex=-1, т.е. ничего не выбрано.

1. Поместите на форму кнопку Связь. В OnClick этой кнопки напишите код:

Query1. DataSource:=DataSource1; (это надо для последующего объединения

таблиц, т.е. мы указываем источник главной таблицы)

// выключение режима редактирования

if Query1.RequestLive then begin

Query1.Active:=false;

Query1.RequestLive:=False;

Query1.Active:=true;

end;

// выключение радиокнопок

RadioGroup1.ItemIndex:=-1;

// объединение таблиц

Query1.Close;

Query1.SQL.Clear;

Query1.SQL.Add('select \* from grp2 where NFcl=:NF');

Query1.Open;