

## ЛАБОРАТОРНАЯ РАБОТА №5. Дополнительные возможности СУБД MS ACCESS

Цель: изучить среду проектирования баз данных в MS Access.

### Порядок выполнения лабораторной работы:

1. Изучить методические указания к лабораторной работе.
2. Обработать ошибку «Отсутствие в списке».
3. Добавить для форм обработку событий «Открытие» и «Закрытие».
4. Создать группу переключателей.
5. Организовать поиск в форме.
6. Поменять параметры запуска БД.
7. Оформить комплексный отчет по 3-5 лабораторным работам.

Требования к отчету:

- Титульный лист.
- Цель лабораторных работ.
- Принтскрины этапов 3-5 лабораторных работ.
- Выводы по лабораторным работам.

8. Защита лабораторной работы (только при наличии печатного отчета):
  - любой вопрос по выполнению лабораторной работы;
  - любой вопрос по отчету;
  - любой вопрос из контрольных вопросов.

### Пояснения по выполнению лабораторной работы

1. В созданной БД создать форму для изменения и добавления записей одновременно в нескольких таблицах.

Для этого нужно обработать ошибку «отсутствие в списке»: создать процедуру, позволяющую не ограничиваться существующим списком для полей со списком. При вводе отсутствующего в списке значения:

- должно выводиться окно сообщения об ошибке;
- должна открываться вспомогательная форма для добавления новой записи в таблицу, данные из которой содержатся в поле со списком;
- после закрытия вспомогательной формы в выпадающем списке должно присутствовать новое значение.

2. В режиме конструктора форм и отчетов посмотреть, где прописывается информация о том, *откуда* берутся данные в форме/отчете и в составляющих их элементах.
3. Добавить к формам, вызываемым из Кнопочной формы обработку событий «Открытие», «Закрытие». При открытии вызванной формы, Кнопочная форма должна закрываться, при закрытии формы, открываться обратно. Продемонстрировать 2 варианта реализации: при помощи макросов и Visual Basic.
4. Создать на одной из форм группу переключателей.  
Переключатели использовать как облегчение ввода данных (результат выбора должен записываться в таблицу) или как фильтр при выводе данных.

5. Добавить поиск информации к форме.

К существующей форме добавить несвязное поле для ввода строки поиска.

Создать новый запрос, с источником данных как на форме, добавить к запросу условие на выбранное поле Like [Forms]![имя\_формы]![имя\_поля\_поиска] & "\*".

Поменять источник данных формы на созданный запрос.

Полю поиска присвоить значение по умолчанию равное «\*» и добавить динамическое обновление источника данных при вводе поисковой строки - обработка события «Изменение».

6. Поменять параметры запуска так, чтобы база запускалась с созданной ранее кнопочной формы без окна базы данных.

\*Параметры запуска можно обойти, чтобы получить полный доступ к файлу Microsoft Access. Для этого при открытии базы данных удерживайте в нажатом положении клавишу SHIFT.

**Пример процедуры обработки события «отсутствие в списке»:**

```
Private Sub <имя_списка>_NotInList(NewData As String, Response As Integer)
```

```
Dim stDocName As String
```

```
    ' окно сообщения
```

```
    If MsgBox("текст в окне", vbYesNo, "заголовок окна") = vbYes Then
```

```
        stDocName = "имя_открываемой_формы"
```

```
    ' открываем форму с указанным именем на добавление (последний аргумент)
```

```
    DoCmd.OpenForm stDocName, , , , acFormAdd
```

```
    ' присвоим введенное в список значение соответствующему полю
```

```
    Forms![имя_открываемой_формы]![имя_поля_в_которое_нужно_добавить_данные] =  
        NewData
```

```
    ' сохраняем значение
```

```
    DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
```

```
    ' отменяем последнее действие в исходной форме (ввод текста)
```

```
    Forms![имя_исходной_формы]![ имя_списка].Undo ' обновляем значения списка
```

```
    Forms![ имя_исходной_формы]![ имя_списка].Requery
```

```
Else
```

```
    Forms![имя_исходной_формы]![ имя_списка].Undo
```

```
End If
```

```
    Response = 0 ' обнуляем ошибку
```

```
End Sub
```

**Пример процедуры поиска:**

```
Private Sub <имя_поля>_Change()
```

```
    Me.Requery ' обновляем форму в соответствии с фильтром
```

```
    Me![имя_поля].SetFocus ' возвращаем фокус на поле поиска
```

```
    Me![имя_поля].SelStart = Len(Me!fSearch) ' устанавливаем курсор на конец строки ввода
```

```
End Sub
```

**Контрольные вопросы:**

1. Как можно обработать ошибку «отсутствие в списке»?
2. Где прописывается информация о том, *откуда* берутся данные в форме/отчете?
3. Что такое обработка событий?
4. Какие события можно обработать?
5. Как реализовать обработку событий?
6. Что такое связанные, несвязные и вычисляемые элементы форм?
7. Что такое группа переключателей? Для чего она необходима?
8. Для чего необходимо менять параметры запуска БД?