Structuring A Database

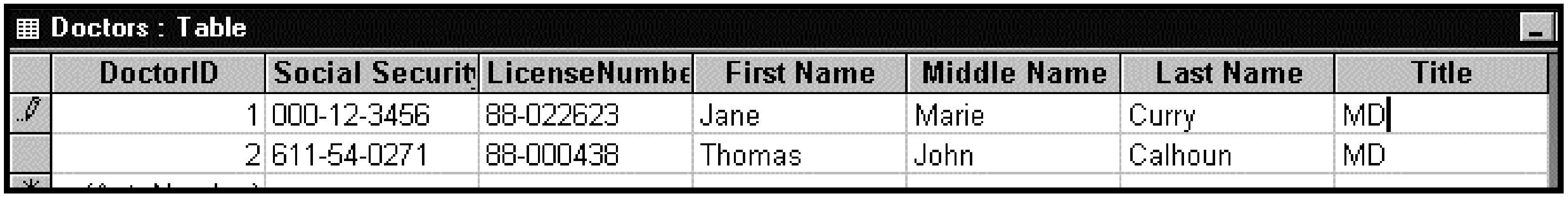
CSE 100

Plopping data into tables haphazardly is a guaranteed way to create consistency problems and limit what can be extracted from the data base. Good database design is a complex subject, but we can learn the basics

FIT 100

Recall ...

- A *database* is a set of *tables*
- The tables are sets of *records* known as *tuples* composed of *fields* each having values of from some primitive data type

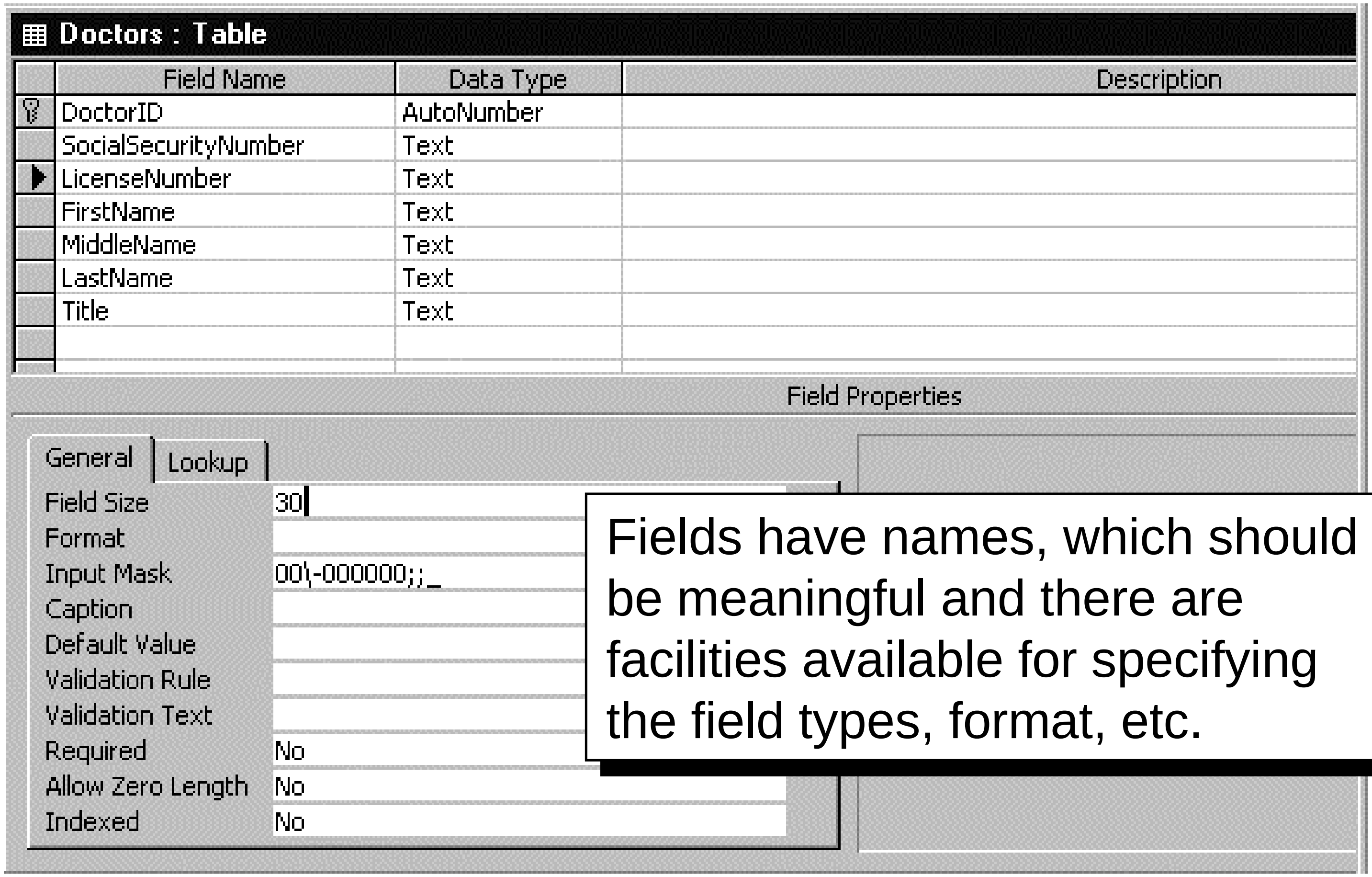


Doctors : Table

	DoctorID	Social Securit	LicenseNumbe	First Name	Middle Name	Last Name	Title
✎	1	000-12-3456	88-022623	Jane	Marie	Curry	MD
	2	611-54-0271	88-000438	Thomas	John	Calhoun	MD
*	(AutoNumber)						

A table named Doctors composed of two tuples or records composed of seven fields, the first of which is an integer and the rest of which are text fields, some with specific formats

Specifying A Table



Doctors : Table

Field Name	Data Type	Description
DoctorID	AutoNumber	
SocialSecurityNumber	Text	
LicenseNumber	Text	
FirstName	Text	
MiddleName	Text	
LastName	Text	
Title	Text	

Field Properties

Field Size	30
Format	
Input Mask	00\-000000;;_
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	No
Indexed	No

Fields have names, which should be meaningful and there are facilities available for specifying the field types, format, etc.

FIT 100 Terminology

- The structure of a data base is called a *database schema*
- The schema specifies …
  - The list of table names forming the database
  - For each table, the fields of its records
  - For each field, its attributes or properties, i.e. data type, key or not key, default value, etc.
- A database as the word is normally used, i.e. tables with specific contents, is known as a database *instance* (of a data base schema)
- There can be many instances of a single database schema

FIT 100

Designing A Database Schema

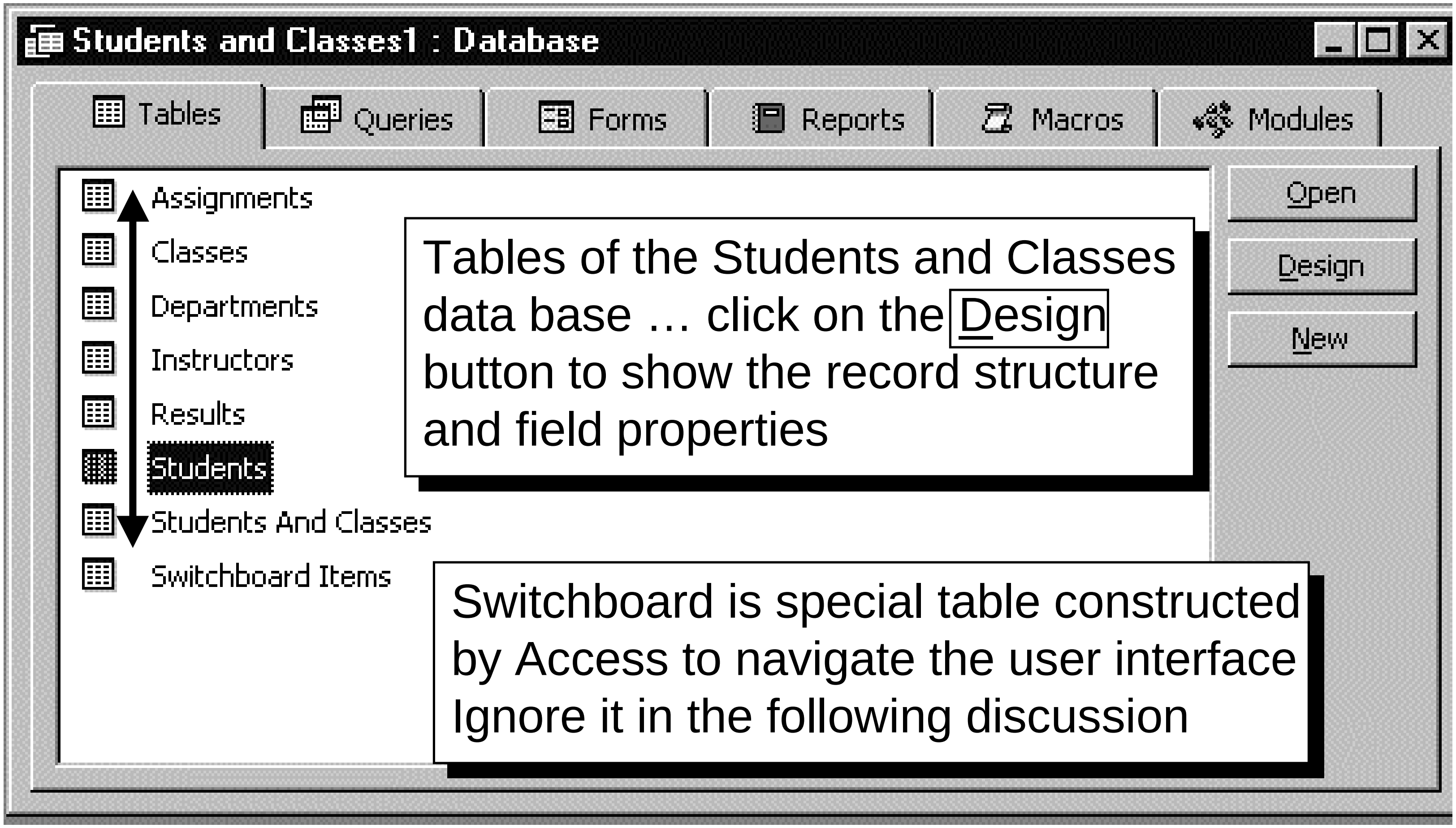
- Suppose a college wants a database of their students, faculty, courses taught, student transcripts, and so forth, what things should go into a design and how should it be organized?
  - Students: first name, last name, home address, transcript …
  - Faculty: first name, last name, SS#, home address, rank …
  - Courses: class name, number, students attending, grades …
- Deciding on the schema is called “database design” and it takes a little study to do right … but it’s easy to see the principles in action

Avoid redundancy!

FIT 100

An Example Data Base Schema

- Consider the Students & Classes DB from Access

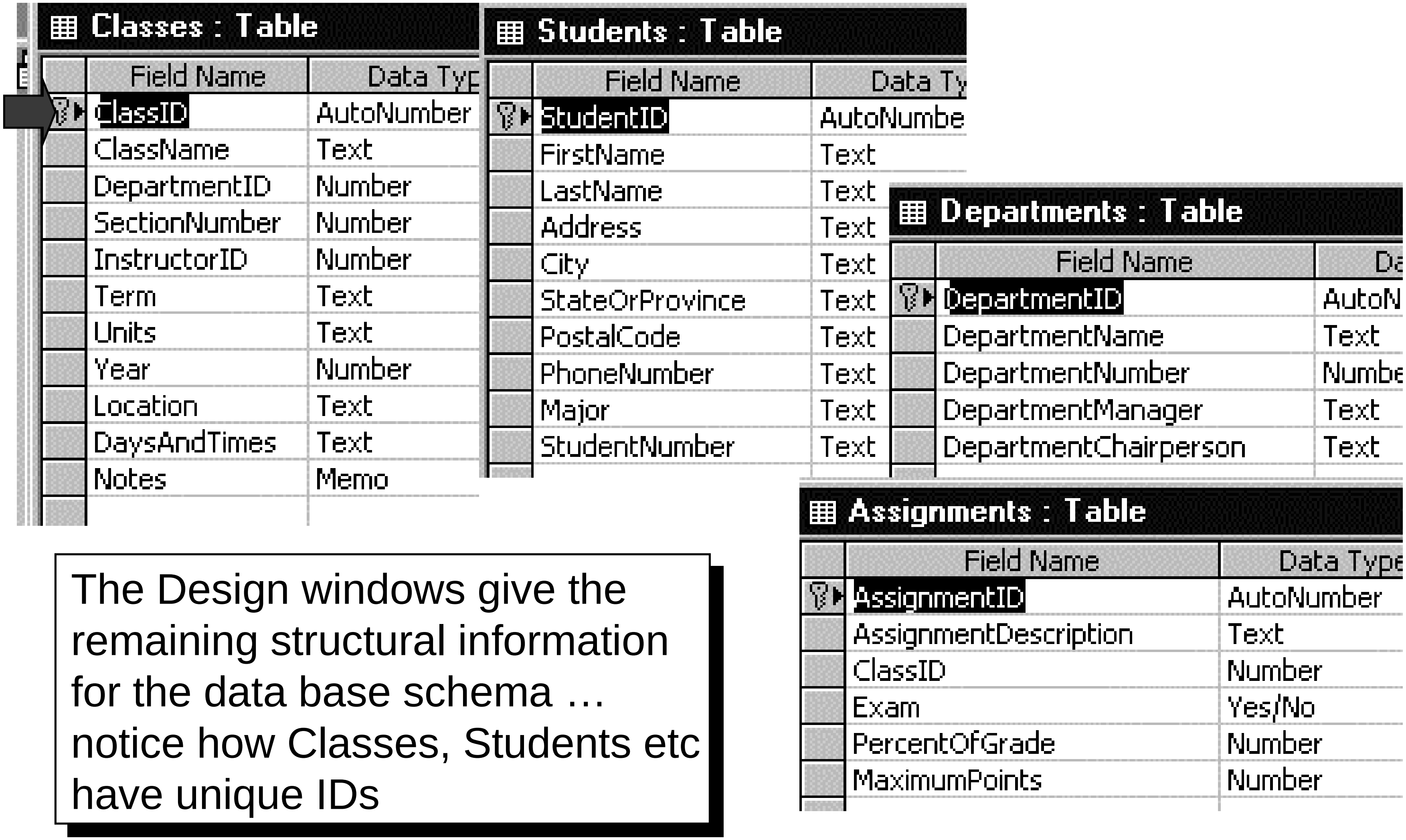


Tables of the Students and Classes data base … click on the Design button to show the record structure and field properties

Switchboard is special table constructed by Access to navigate the user interface Ignore it in the following discussion

FIT 100

More Of The S&C DB Schema



Classes : Table

Field Name	Data Typ
ClassID	AutoNumber
ClassName	Text
DepartmentID	Number
SectionNumber	Number
InstructorID	Number
Term	Text
Units	Text
Year	Number
Location	Text
DaysAndTimes	Text
Notes	Memo

Students : Table

Field Name	Data Ty
StudentID	AutoNumbe
FirstName	Text
LastName	Text
Address	Text
City	Text
StateOrProvince	Text
PostalCode	Text
PhoneNumber	Text
Major	Text
StudentNumber	Text

Departments : Table

Field Name	Da
DepartmentID	AutoN
DepartmentName	Text
DepartmentNumber	Numbe
DepartmentManager	Text
DepartmentChairperson	Text

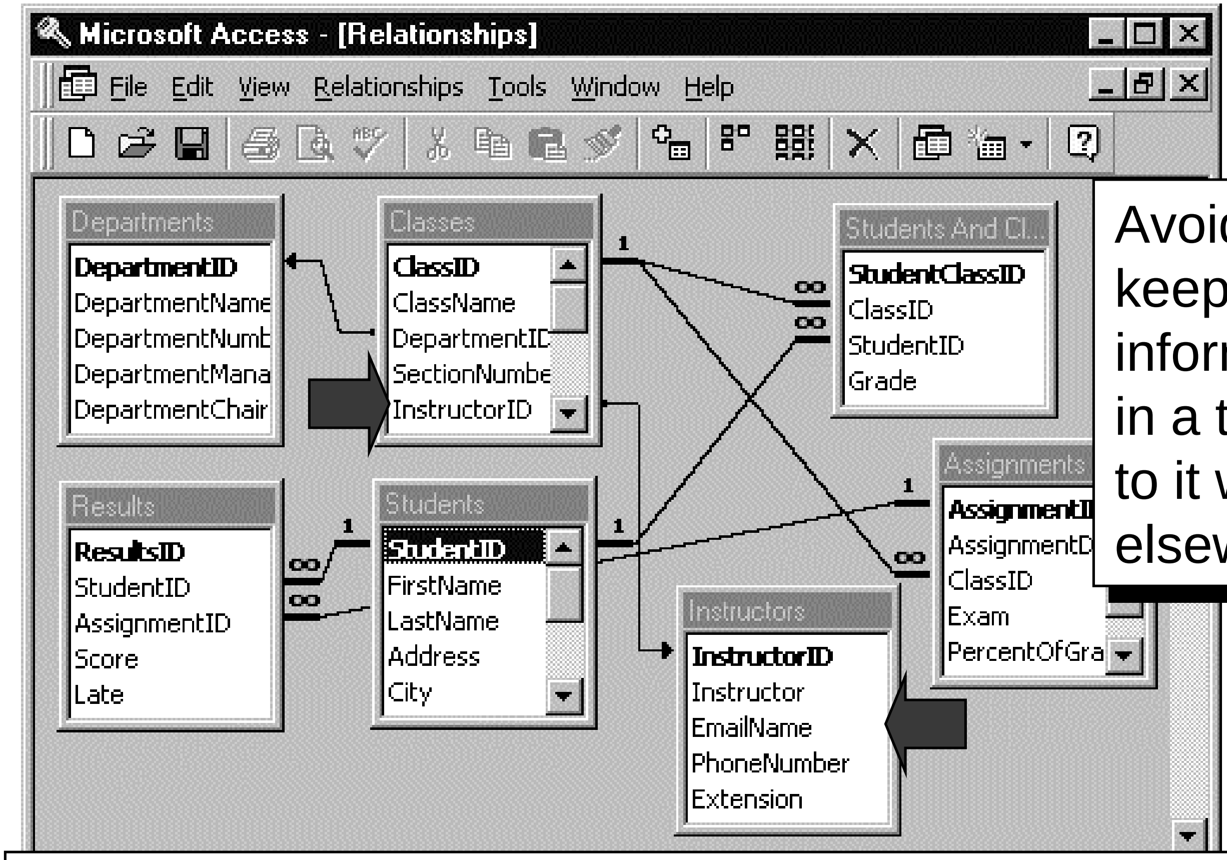
Assignments : Table

Field Name	Data Type
AssignmentID	AutoNumber
AssignmentDescription	Text
ClassID	Number
Exam	Yes/No
PercentOfGrade	Number
MaximumPoints	Number

The Design windows give the remaining structural information for the data base schema … notice how Classes, Students etc have unique IDs

FIT 100

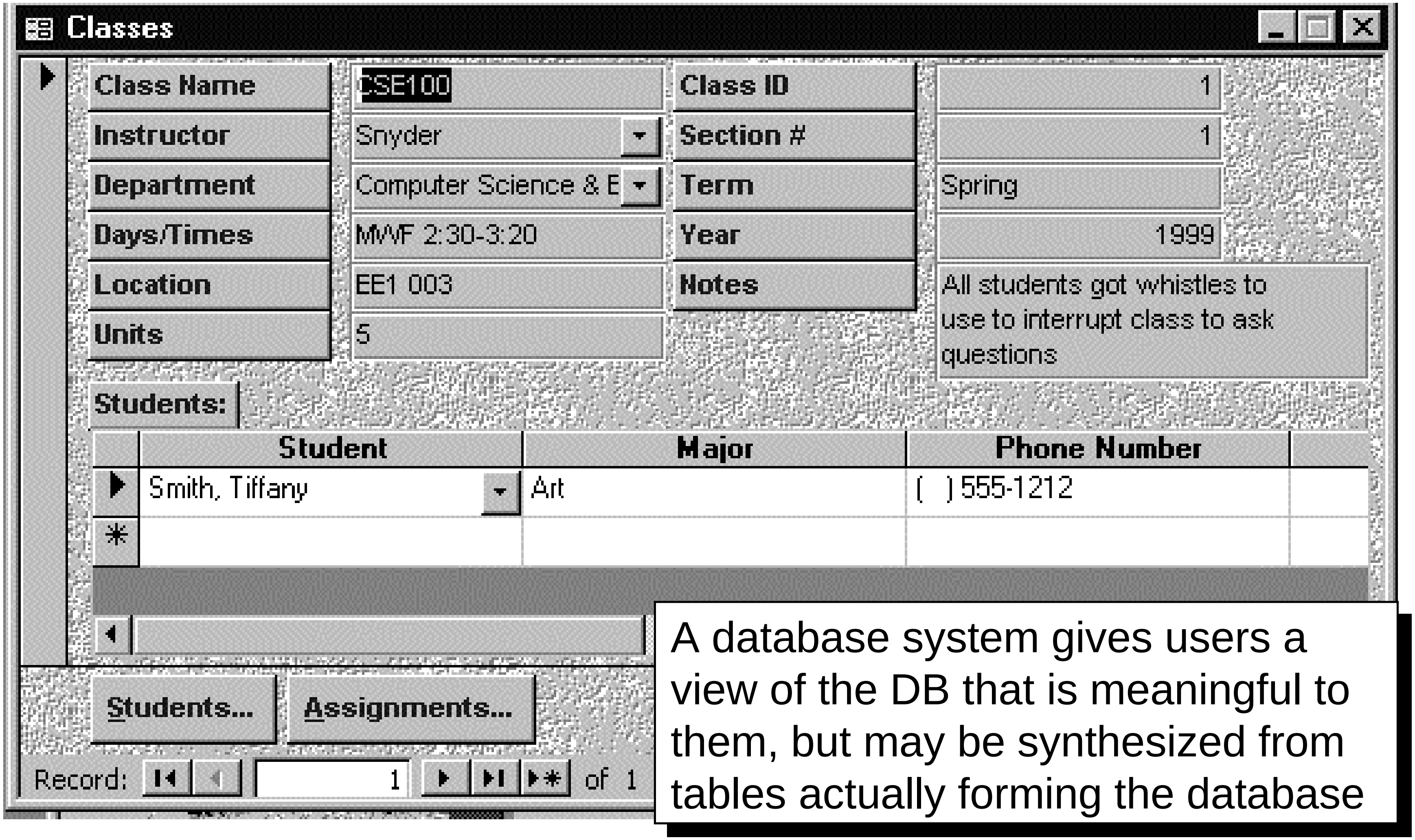
The Tables For The University



Avoid redundancy by keeping like information together in a table and linking to it when it is needed elsewhere

Bad design … in the Classes table, do not include instructor's name, email, etc … rather link to the instructor information by ID

The User's View



Class Name	CSE100	Class ID	1
Instructor	Snyder	Section #	1
Department	Computer Science & E	Term	Spring
Days/Times	MWF 2:30-3:20	Year	1999
Location	EE1 003	Notes	All students got whistles to use to interrupt class to ask questions
Units	5		

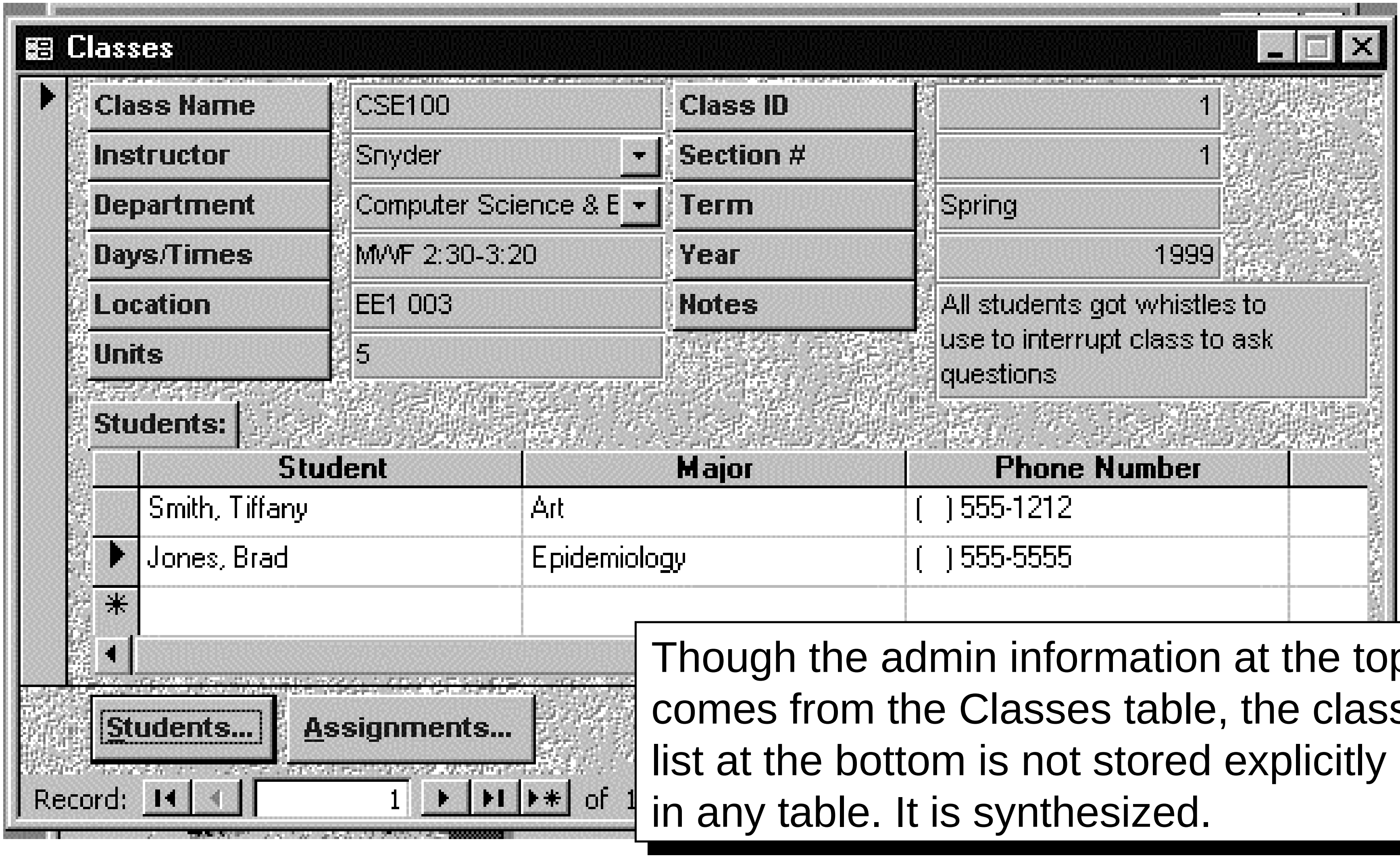
Students:

Student	Major	Phone Number
Smith, Tiffany	Art	( ) 555-1212

A database system gives users a view of the DB that is meaningful to them, but may be synthesized from tables actually forming the database

FIT 100

Consider A Student’s List Of Classes



Class Name	CSE100	Class ID	1
Instructor	Snyder	Section #	1
Department	Computer Science & E	Term	Spring
Days/Times	MWF 2:30-3:20	Year	1999
Location	EE1 003	Notes	All students got whistles to use to interrupt class to ask questions
Units	5		

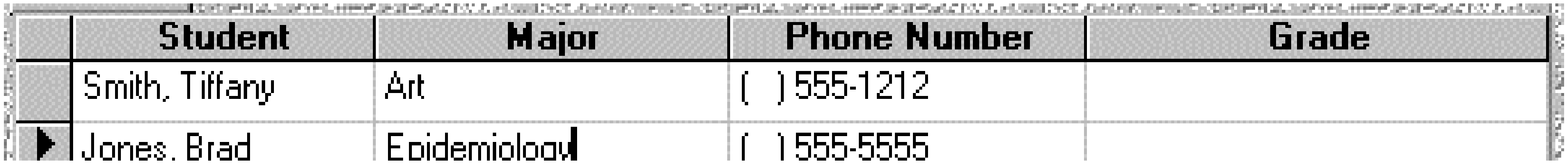
Students:

Student	Major	Phone Number
Smith, Tiffany	Art	( ) 555-1212
Jones, Brad	Epidemiology	( ) 555-5555

Though the admin information at the top comes from the Classes table, the class list at the bottom is not stored explicitly in any table. It is synthesized.

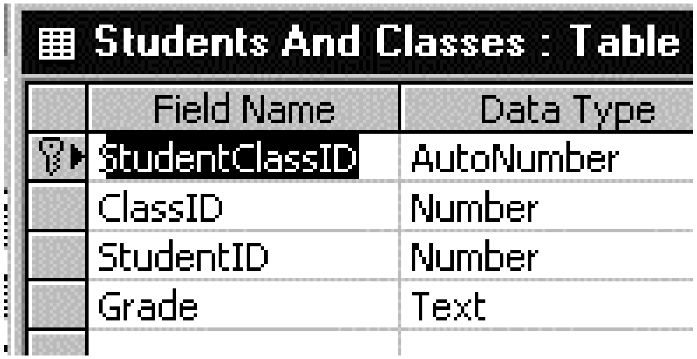
FIT 100

Synthesizing The Class List



	Student	Major	Phone Number	Grade
	Smith, Tiffany	Art	( ) 555-1212	
▶	Jones, Brad	Epidemiology	( ) 555-5555	

- One table -- Students And Classes -- contains records that associate students with classes



Students And Classes : Table

	Field Name	Data Type
▶	StudentClassID	AutoNumber
	ClassID	Number
	StudentID	Number
	Grade	Text

Each tuple is set up by STAR when you register

- By listing all records with FIT100’s *ClassID*, a table is created of the students in FIT100 by *StudentID*
- By looking up each student using *StudentID*, the other fields of the class list can be located

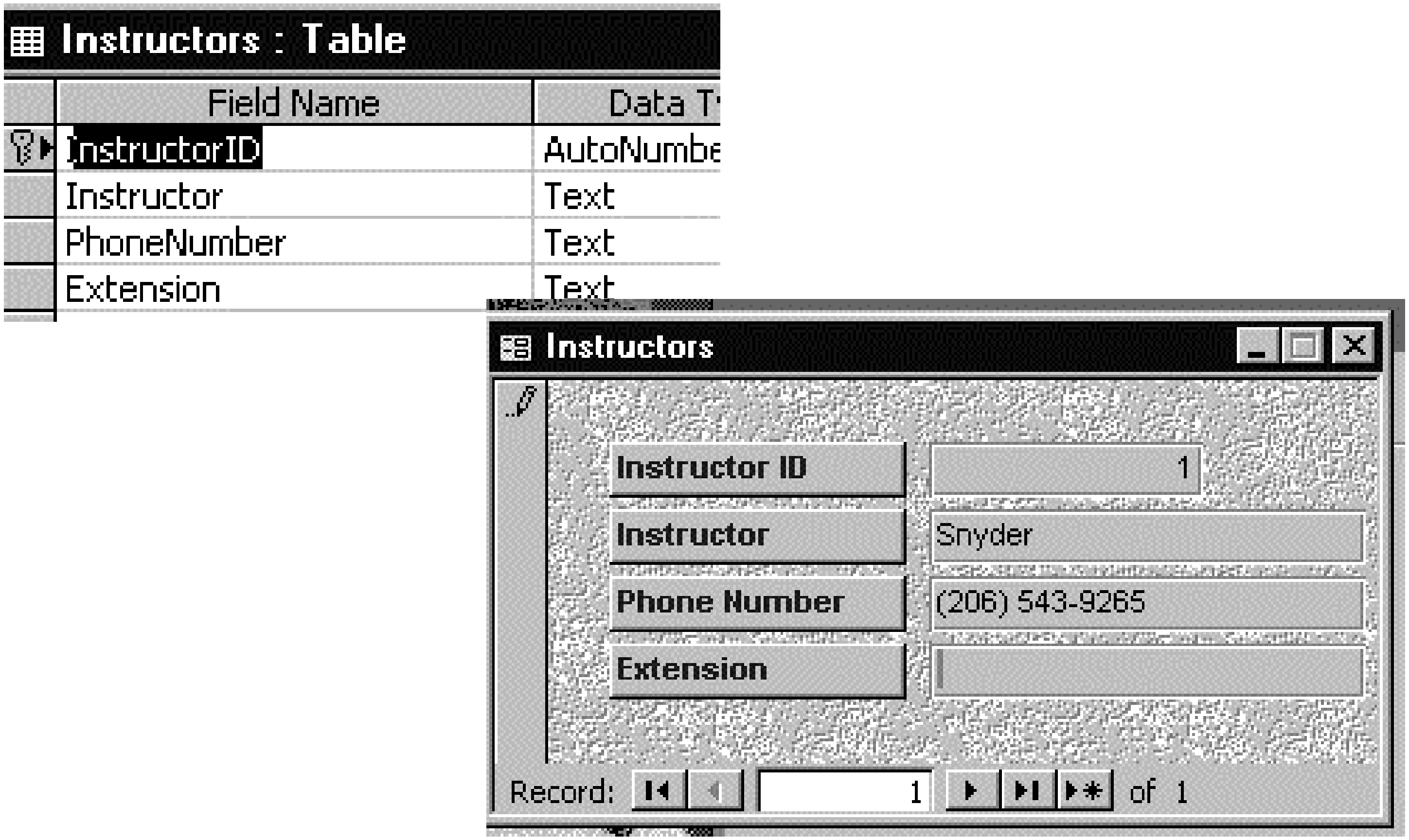
Why Use This Schema?

- Associating a student with a class is the logical idea behind *registering* for a class, so Students & Classes corresponds to a real phenomenon -- a plus
- Having classes listed in the student record violates the goal of a fixed length record, and makes it cumbersome to create a class list -- minuses
- Having students listed in the class record violates the fixed length record goal, and makes it cumbersome to create a registration list for each student -- minus
- “Registering students” -- what STAR does -- can be done without touching either Students or Classes tables -- a plus

FIT 100

Not All Views Are Synthesized

- Many tables will be of interest on their own, too



Instructors : Table

Field Name	Data T
InstructorID	AutoNumbe
Instructor	Text
PhoneNumber	Text
Extension	Text

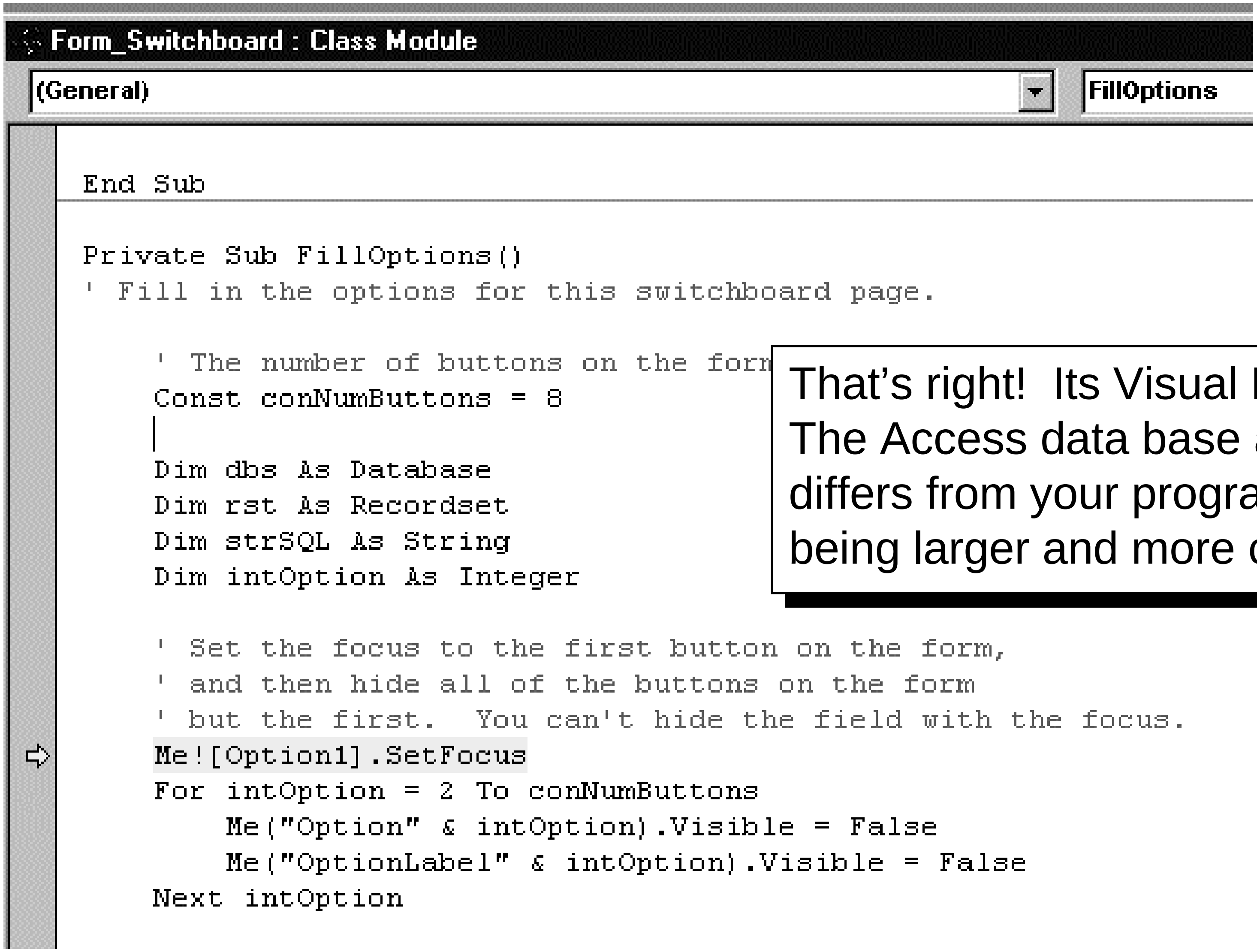
Instructors

Instructor ID	1
Instructor	Snyder
Phone Number	(206) 543-9265
Extension	

Record: 1 of 1

FIT 100

How Was Access Created?



```
End Sub

Private Sub FillOptions()
' Fill in the options for this switchboard page.

    ' The number of buttons on the form
    Const conNumButtons = 8

    Dim dbs As Database
    Dim rst As Recordset
    Dim strSQL As String
    Dim intOption As Integer

    ' Set the focus to the first button on the form,
    ' and then hide all of the buttons on the form
    ' but the first.  You can't hide the field with the focus.
    Me![Option1].SetFocus
    For intOption = 2 To conNumButtons
        Me("Option" & intOption).Visible = False
        Me("OptionLabel" & intOption).Visible = False
    Next intOption
```

That’s right! Its Visual Basic … The Access data base application differs from your programming by being larger and more complex.