Databases 1 – Organisation and Creation

Lecture 18 – COMPSCI111/111G SS 2016

Today's lecture

- What is a database?
- Understanding how data is organised in a database
- Creating a database in Microsoft Access

What is a database?

- A (typically large) collection of data about a particular topic, organized systematically
- Examples:
  - Catalogue of library books
  - Patients' files in a clinic
  - Entries in an address book
  - Students in a class
- Computers allow us to store and manage databases that contain very large amounts of information

Aspects of a database

- Before we can create our database, we need to decide how to:
  1. **Organize** data in our database
  2. **Enter** data in our database
  3. **Retrieve** data from our database
  4. **Present** the retrieved data to the user

1. Organising data - models

- A **model** defines how data is organized and structured within the database
  - We're going to look at the relational model in this course
- When deciding what data to store in a database, we need to think about:
  - **Entities:** things about which we store information
    - Eg. students in uni, courses in uni
  - **Relationships:** specific connections among entities
    - Eg. *students* enrolled in *CompSci111/111G*

1. Organising data - tables

- The **relational model** was developed by Edgar Codd in 1970
- Data is stored and organized in tables
  - A table's columns are called **fields**; an entity's attributes
  - A table's rows are called **records**; one instance of an entity
- A collection of tables form a **database**

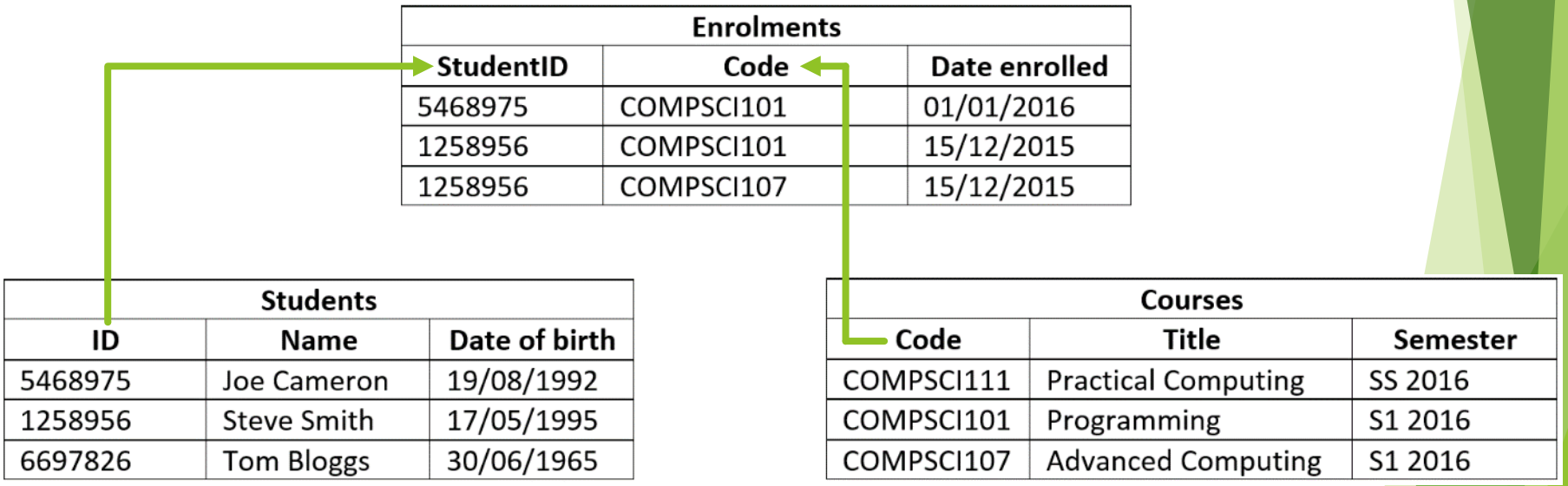
Field

StudentId	Name	Address	Phone
12345	C. Brown	12 Apple St.	555-1234
67890	L. Van Pelt	34 Pear Ave.	555-5678
22222	P. Patty	56 Grape Blvd.	555-9999

Record

1. Organising data

- Tables are connected together using **relationships**, thereby creating connections between different entities



Enrolments		
StudentID	Code	Date enrolled
5468975	COMPSCI101	01/01/2016
1258956	COMPSCI101	15/12/2015
1258956	COMPSCI107	15/12/2015

Students		
ID	Name	Date of birth
5468975	Joe Cameron	19/08/1992
1258956	Steve Smith	17/05/1995
6697826	Tom Bloggs	30/06/1965

Courses		
Code	Title	Semester
COMPSCI111	Practical Computing	SS 2016
COMPSCI101	Programming	S1 2016
COMPSCI107	Advanced Computing	S1 2016

1. Organising data

- There are two parts to a relationship; **primary key** and **foreign key**
- 1. Primary key:
  - Generally, all tables must have a primary key field
  - All records must have a value in the primary key field
  - The primary key's value must be unique

Primary key

Field Name	Data Type
ID	AutoNumber
First Name	Text
E-mail Address	Text
Level	Text
Room	Text
Date of Birth	Date/Time

1. Organising data

- 2. Foreign key
  - A field in one table that is related to a primary key field in another table
  - Creates a connection between the two fields
  - Can take blank values and/or repeated value depending on the relationship

Enrolments		
StudentID	**Code**	**Date enrolled**
5468975	COMPSCI101	01/01/2016
1258956	COMPSCI101	15/12/2015
1258956	COMPSCI107	15/12/2015

Students		
ID	**Name**	**Date of birth**
5468975	Joe Cameron	19/08/1992
1258956	Steve Smith	17/05/1995
6697826	Tom Bloggs	30/06/1965

Referential integrity

- An important concept underlying relationships between tables
- Referential integrity requires all values of a foreign key field to be:
  - Present in the related primary key field, OR
  - Null (ie. blank)

Referential integrity

Students		
ID	Name	Date of birth
5468975	Joe Cameron	19/08/1992
1258956	Steve Smith	17/05/1995
6697826	Tom Bloggs	30/06/1965

Enrolments		
StudentID	Code	Date enrolled
5468975	COMPSCI101	01/01/2016
1258956	COMPSCI101	15/12/2015
1258956	COMPSCI107	15/12/2015

Insert 9998881, COMPSCI111, 22/12/2015 *into* Enrolments ✗

Insert 6697826, COMPSCI105, 16/12/2015 *into* Enrolments ✓

Insert , COMPSCI101, 01/12/2015 *into* Enrolments ✓

Types of relationships

- There are three kinds of relationship that can exist between tables
- **One to one:** one record in PK related to one record in FK
  - Eg. student can only have one transcript
- **One to many:** one record in PK related to multiple records in FK
  - Eg. student can have multiple emergency contacts
- **Many to many:** multiple records in PK related to multiple records in FK
  - Eg. many students can be enrolled in many papers

Aspects of a database

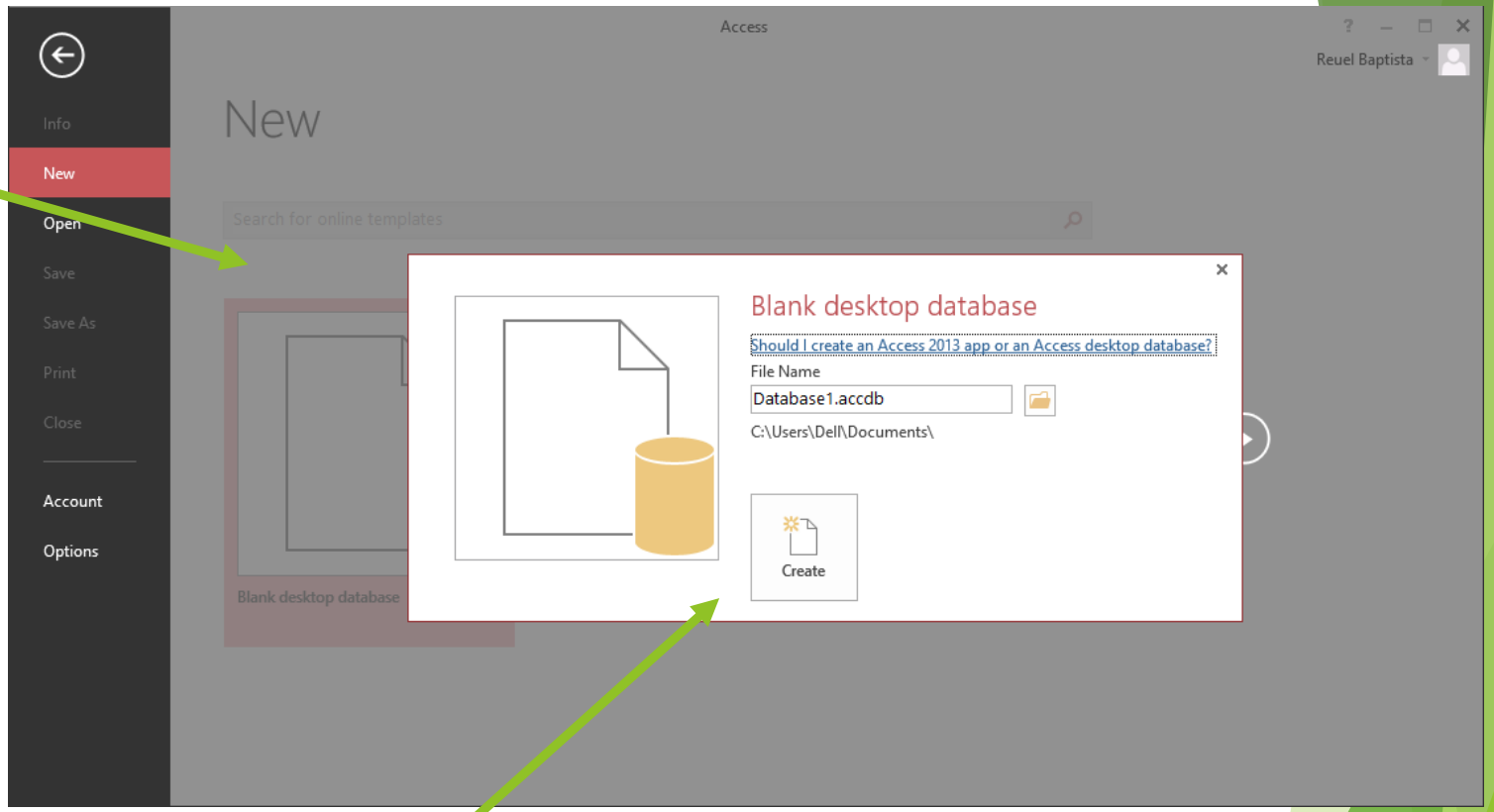
- Before we can create our database, we need to decide how to:
  1. **Organize** data in our database
     - Models, tables, relationships
  2. **Enter** data in our database
  3. **Retrieve** data from our database
  4. **Present** the retrieved data to the user

Database Management System (DBMS)

- Application software that is used to manage databases.
- Four main functions:
  - Definition
  - Update
  - Querying
  - Administration
- Examples:
  - Microsoft Access
  - Microsoft SQL Server

Creating a database

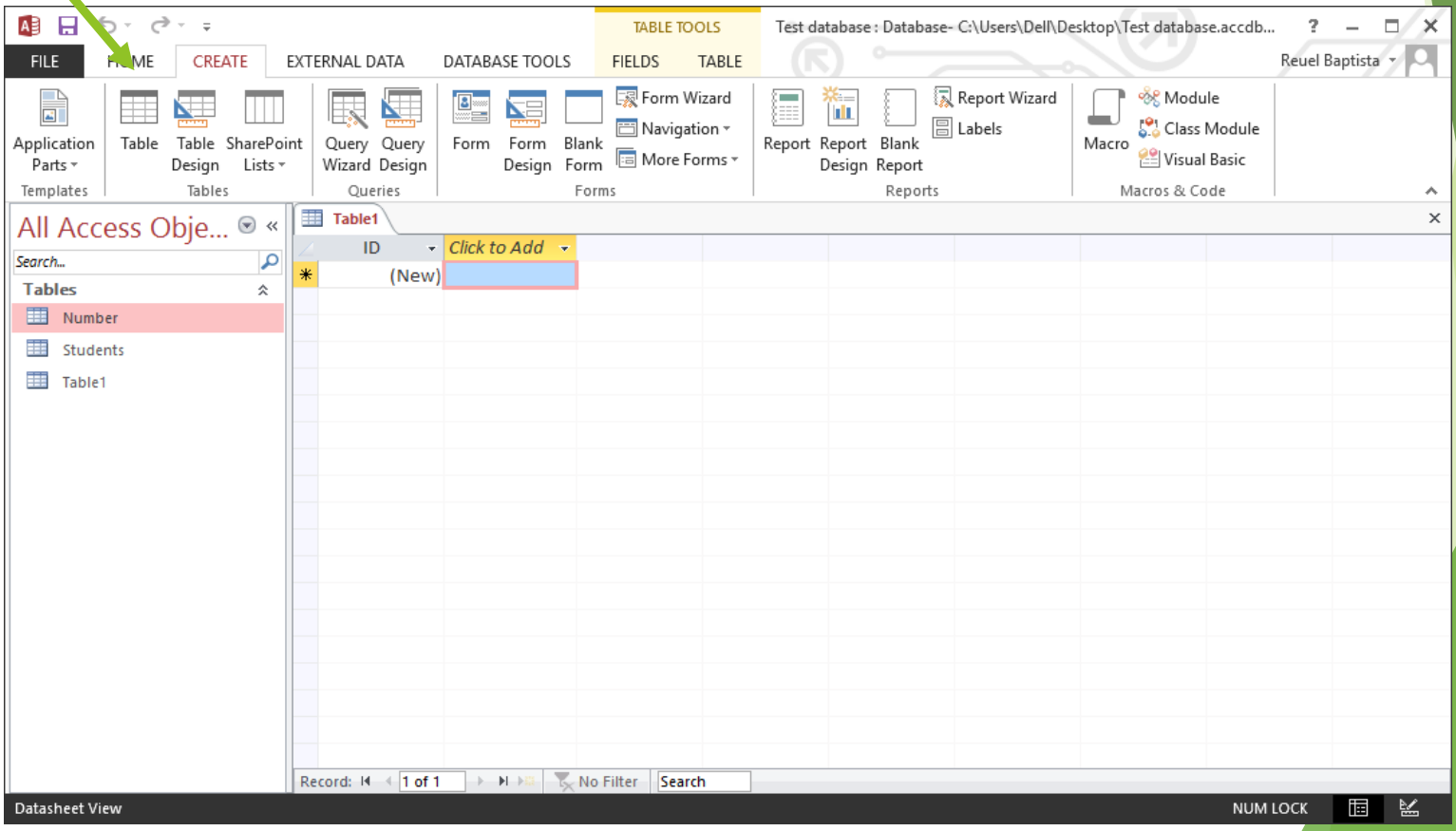
Templates



Creating a new database

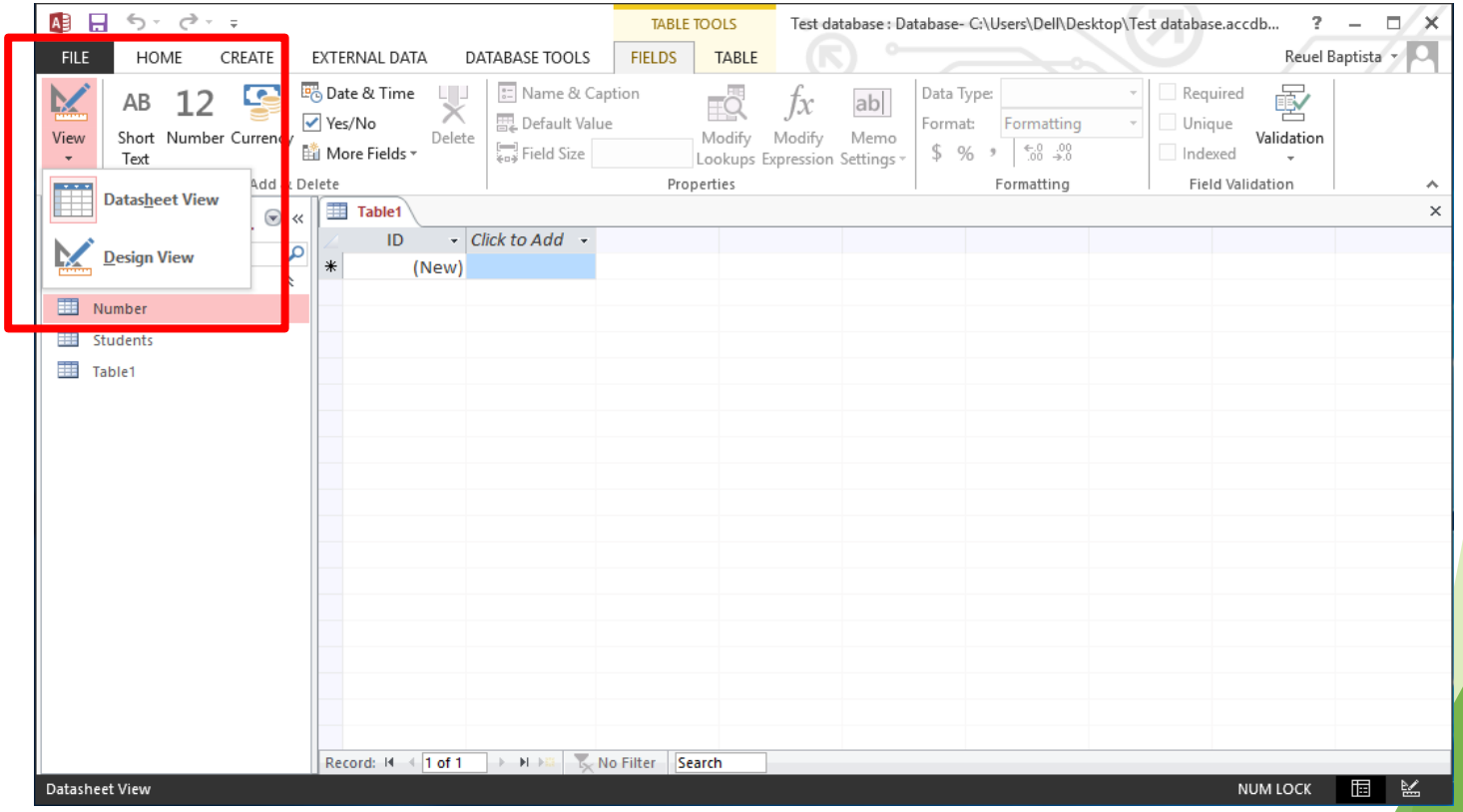
Creating a table

New table button



Creating a table

- **Design view:** create/view the fields in the table
- **Datasheet view:** create/view data in the table



Design view

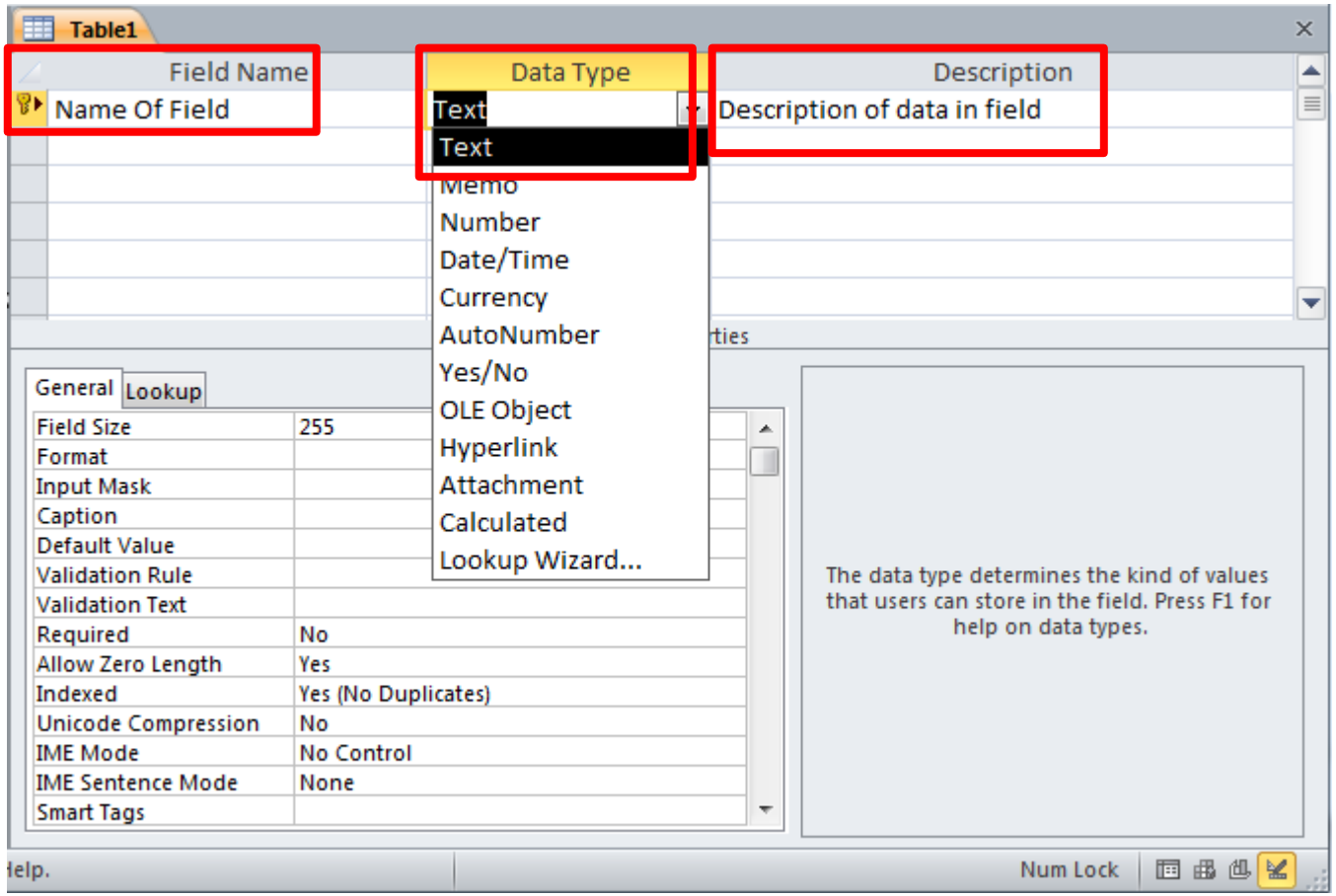


Table1
Field Name
Data Type
Description
Name Of Field
Text
Description of data in field
Text
Memo
Number
Date/Time
Currency
AutoNumber
Yes/No
OLE Object
Hyperlink
Attachment
Calculated
Lookup Wizard...
General
Lookup
Field Size 255
Format
Input Mask
Caption
Default Value
Validation Rule
Validation Text
Required No
Allow Zero Length Yes
Indexed Yes (No Duplicates)
Unicode Compression No
IME Mode No Control
IME Sentence Mode None
Smart Tags
The data type determines the kind of values that users can store in the field. Press F1 for help on data types.
Num Lock

Design view

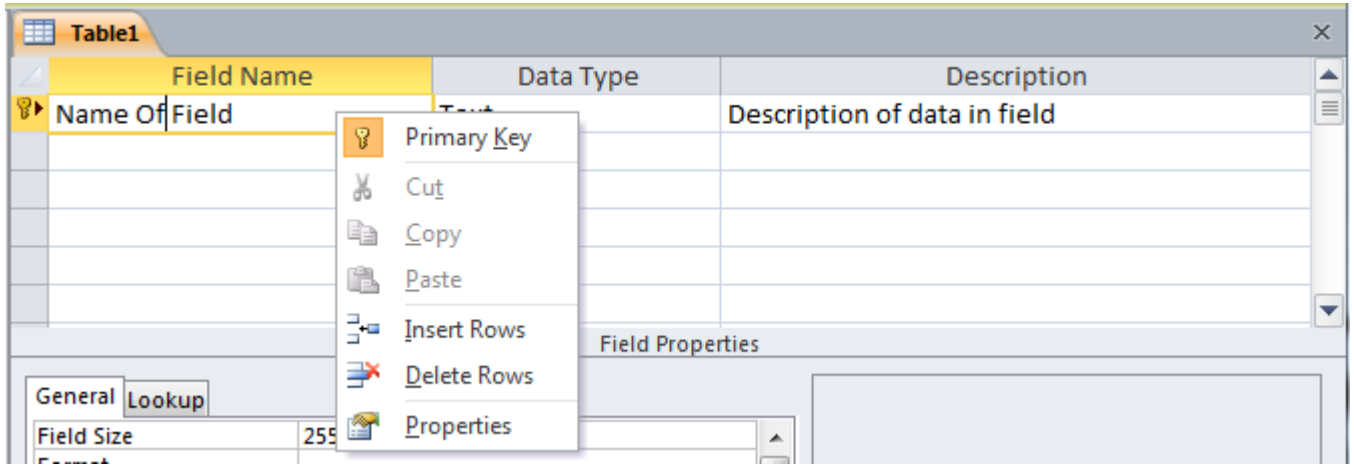
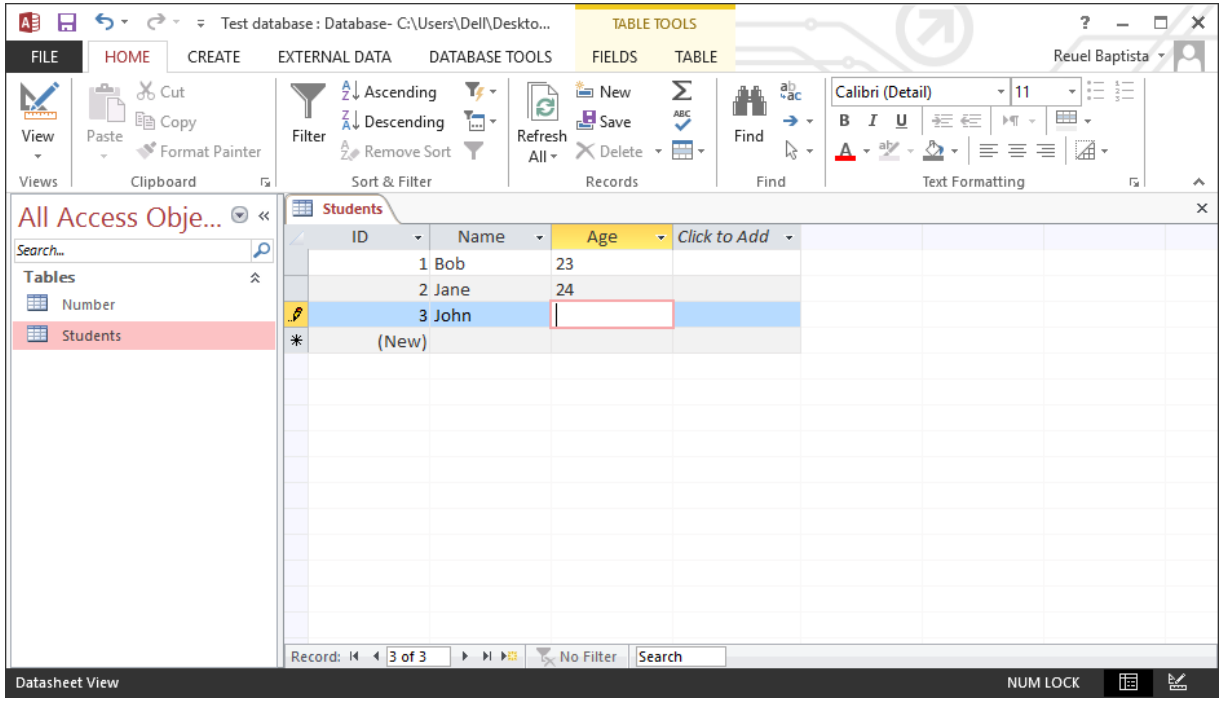


Table1
Field Name
Data Type
Description
Name Of Field
Description of data in field
Primary Key
Cut
Copy
Paste
Insert Rows
Delete Rows
Properties
Field Properties
General
Lookup
Field Size

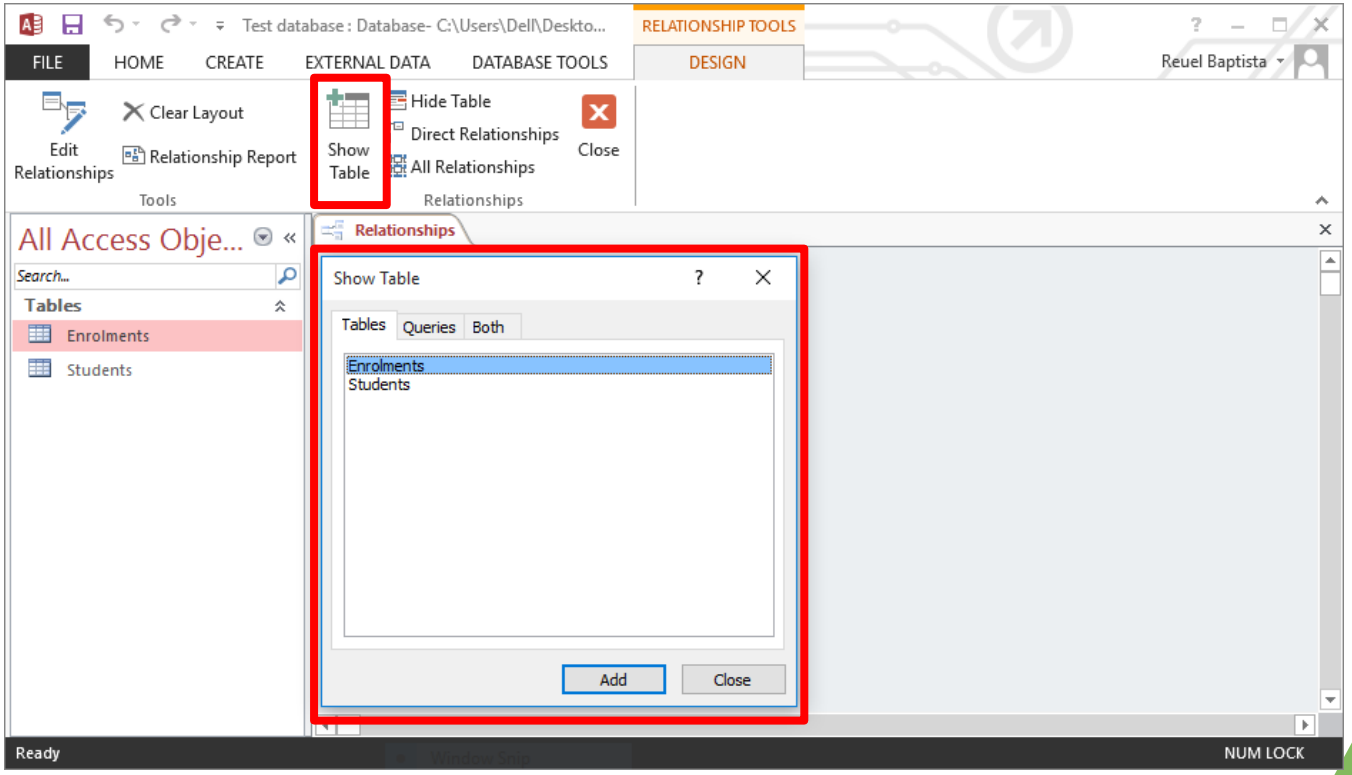
Datasheet view

- Allows us to enter data into our table
- Need to ensure that we enter the correct type of data in each field (eg. no text in a number field)

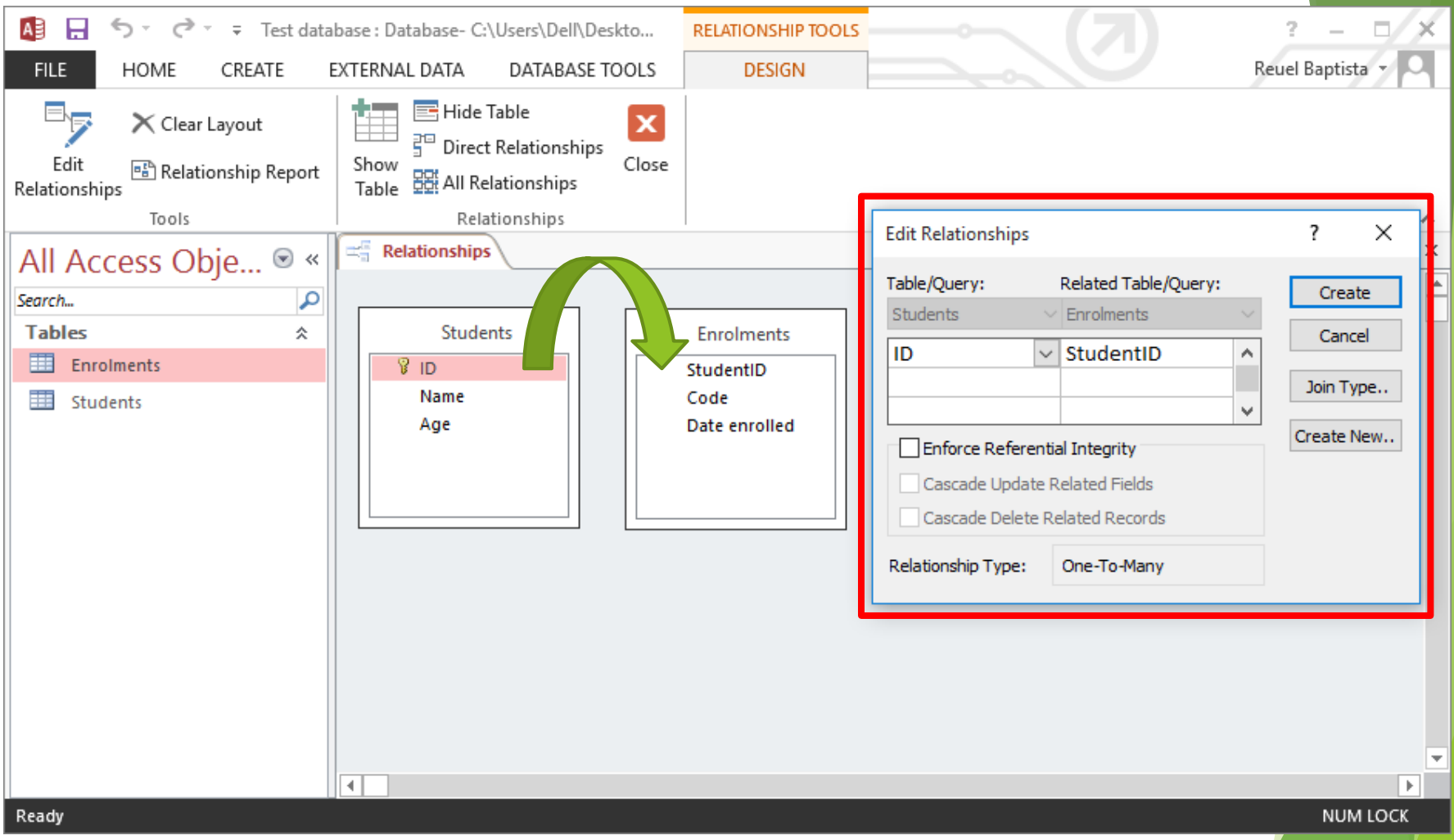


Creating relationships

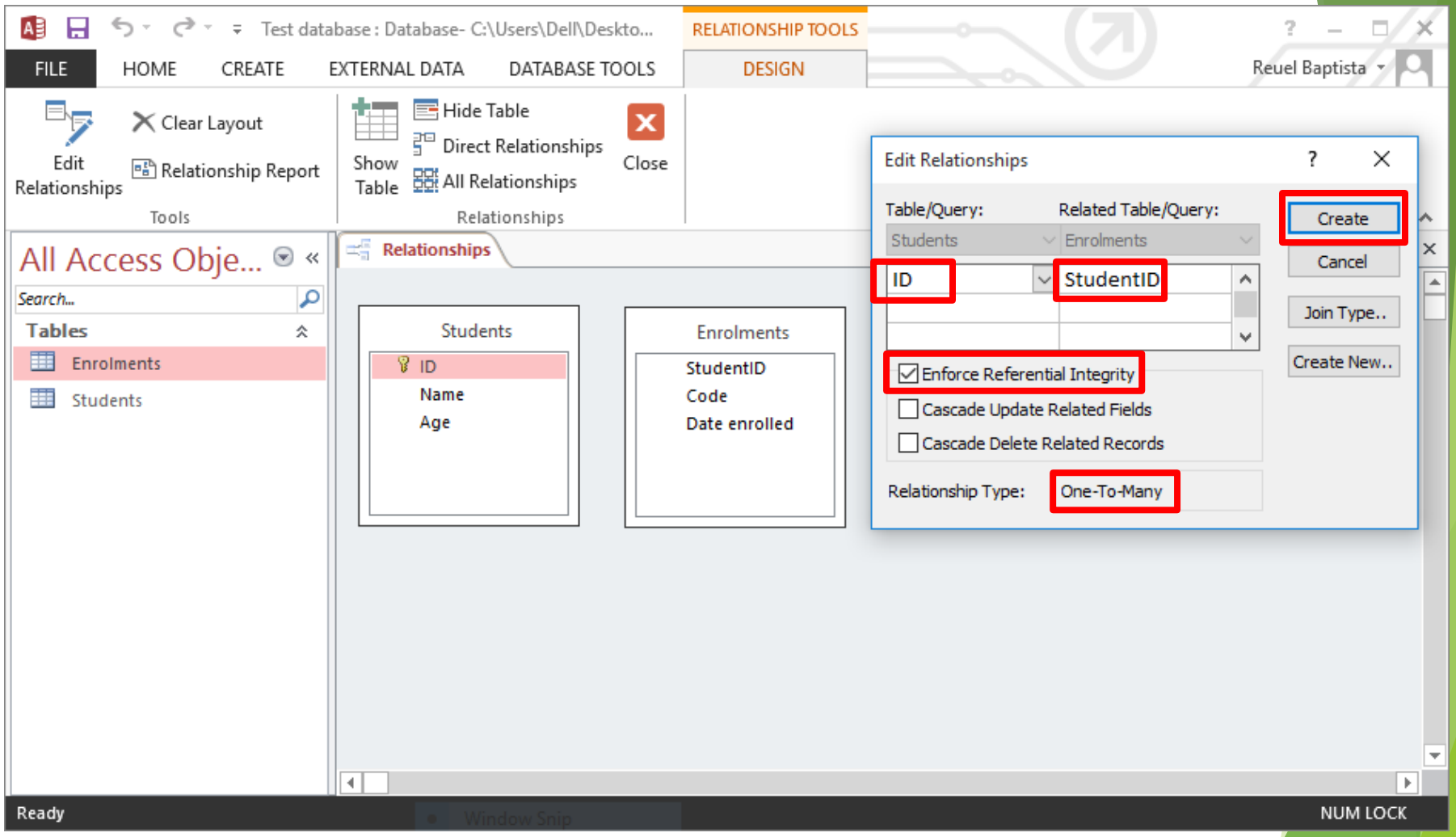
- Relationships view allows us to create relationships between fields in different tables
- Database Tools tab → Relationships button



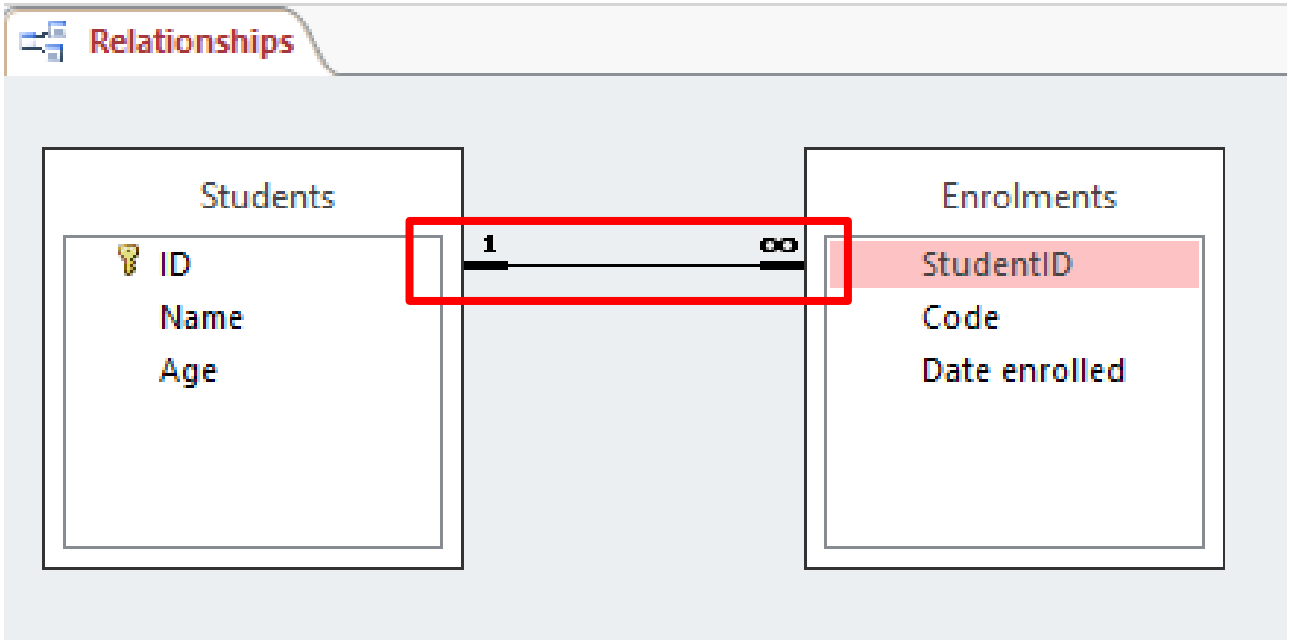
Creating relationships



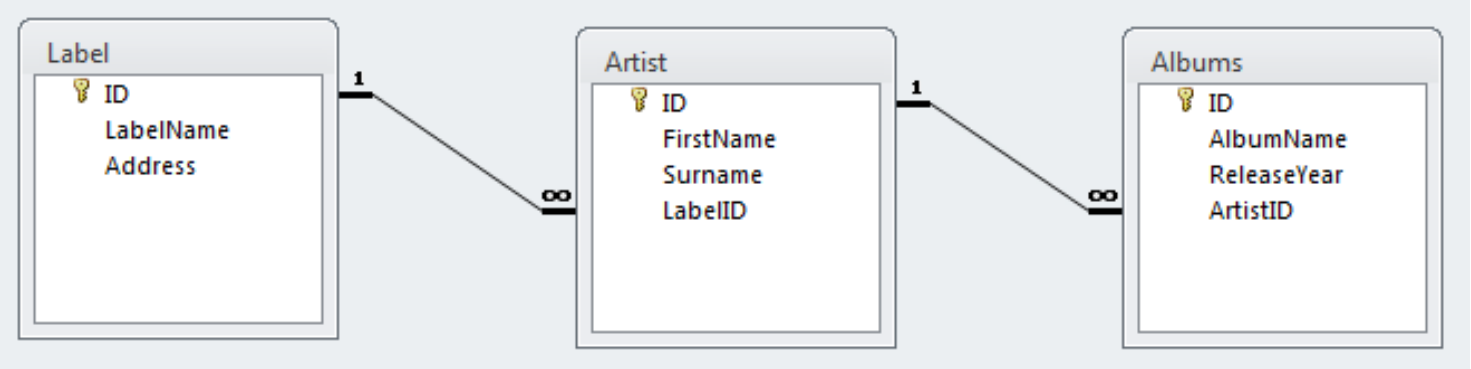
Creating relationships



Creating relationships

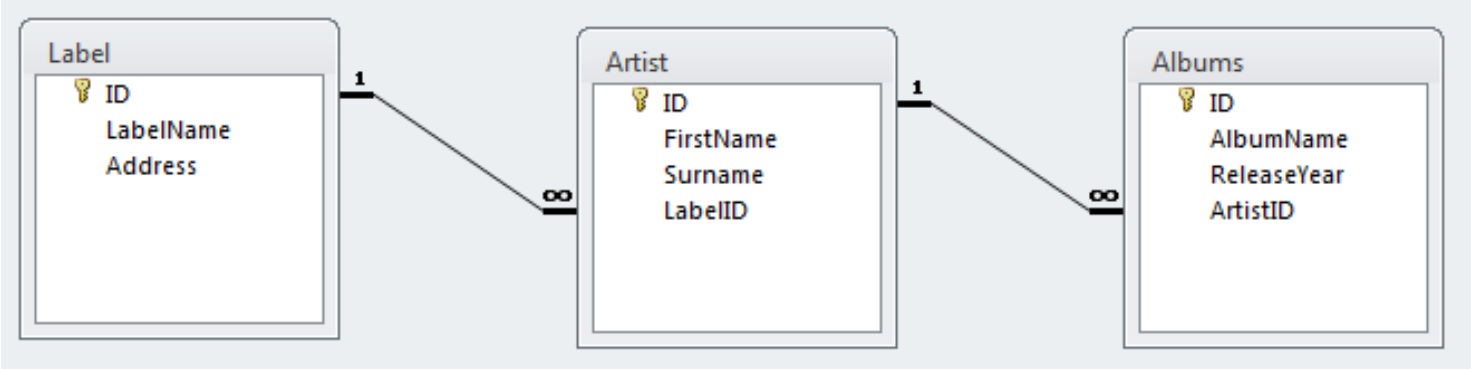


Exercises



1. What is the primary key and the foreign key (if one exists) for the *Label* table?
2. What is the primary key and the foreign key (if one exists) for the *Artist* table?
3. What is the primary key and the foreign key (if one exists) of the *Albums* table?

Answers

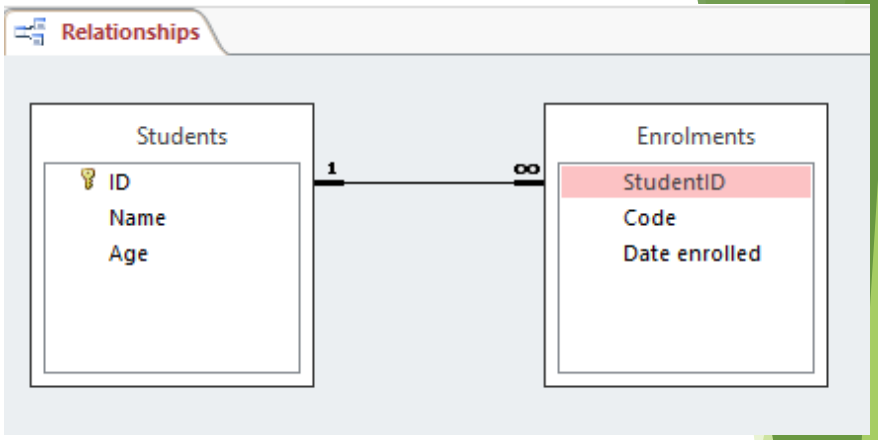


- Label
  - PK: ID
  - FK: none
- Artist
  - PK: ID
  - FK: LabelID
- Albums
  - PK: ID
  - FK ArtistID

Inserting data

Students

ID	Name	Age
1	Bob	23
2	Jane	24
3	John	19

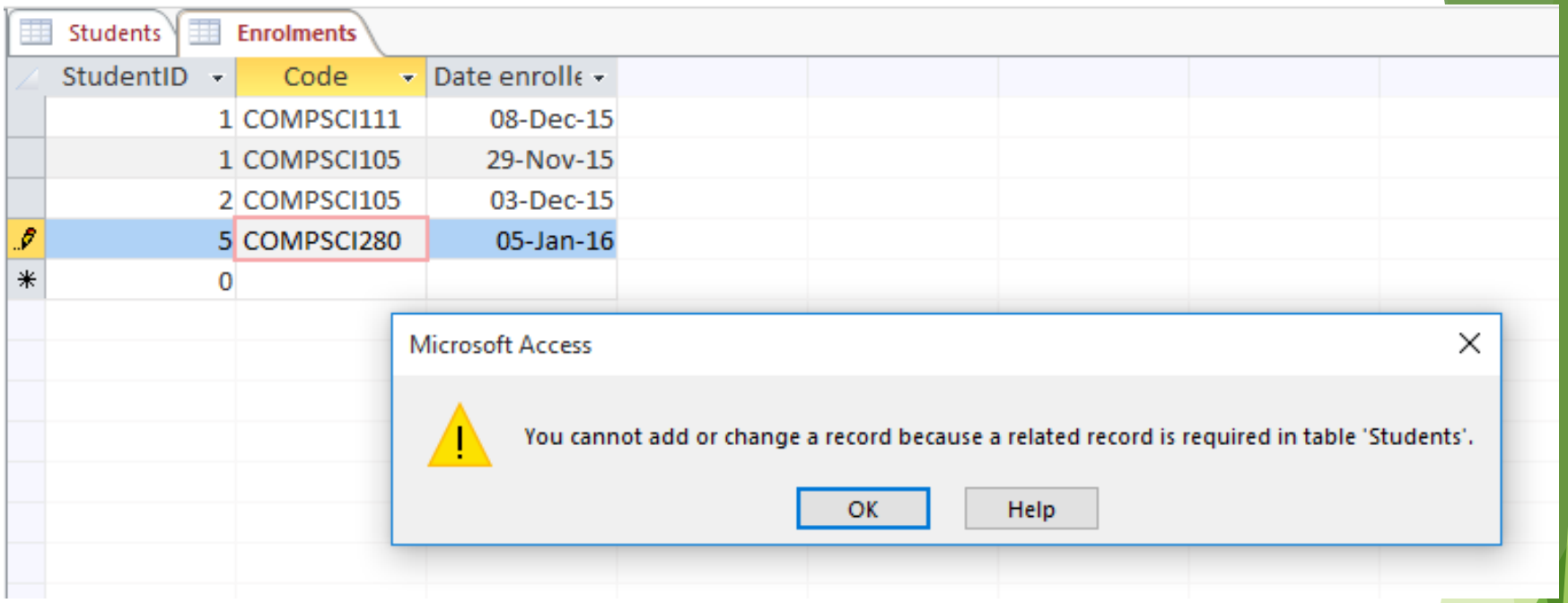


Students | Enrolments

StudentID	Code	Date enrolle
1	COMPSCI111	08-Dec-15
1	COMPSCI105	29-Nov-15
2	COMPSCI105	03-Dec-15
5	COMPSCI280	05-Jan-16
0		

- Can we insert this record in the Enrolments table?

Inserting data



- This won't work; StudentID's value ('5') doesn't exist in the primary key ID

Summary

- A database is used to store information in a systematic and orderly manner
- The relational model uses tables to store information about entities and relationships to connect tables together
- Relationships require tables, primary keys, foreign keys. Referential integrity is an important concept
- Microsoft Access is a popular DBMS that we can use to insert and manage data in our database