

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*

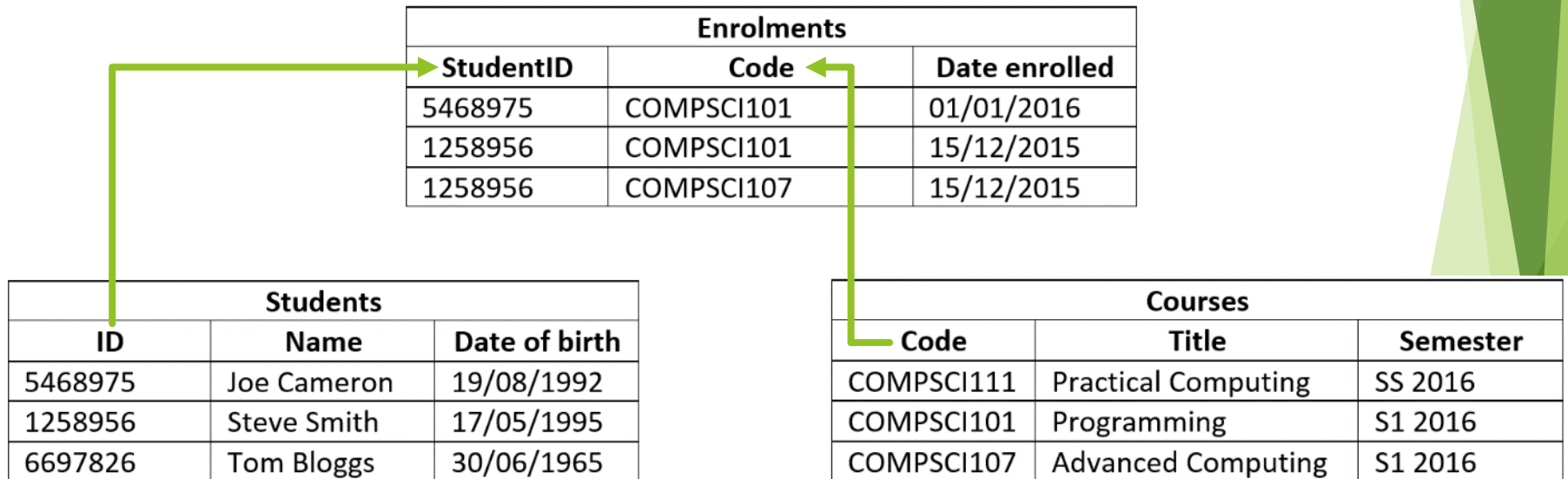
1. Organising data - tables

- ▶ 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
Record	22222	P. Patty	56 Grape Blvd.	555-9999

1. Organising data

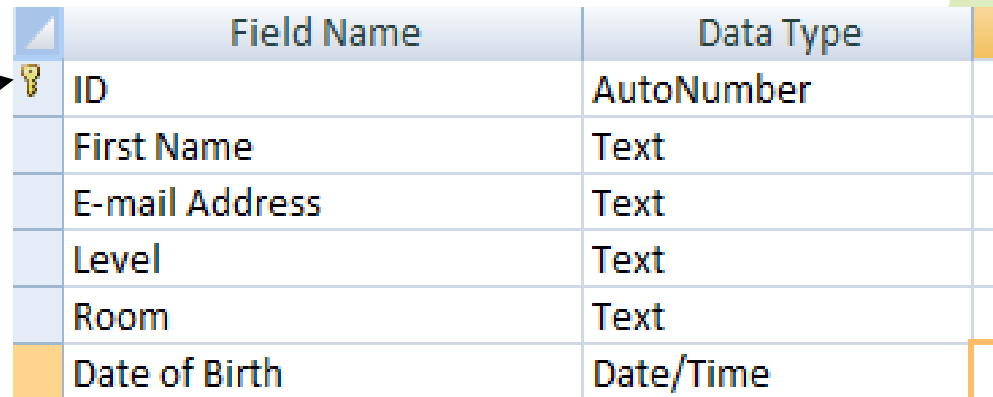
- ▶ Tables are connected together using **relationships**, thereby creating connections between different entities



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

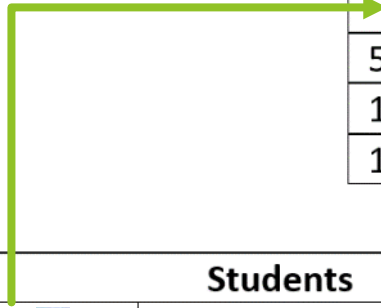
1. Organising data

▶ 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

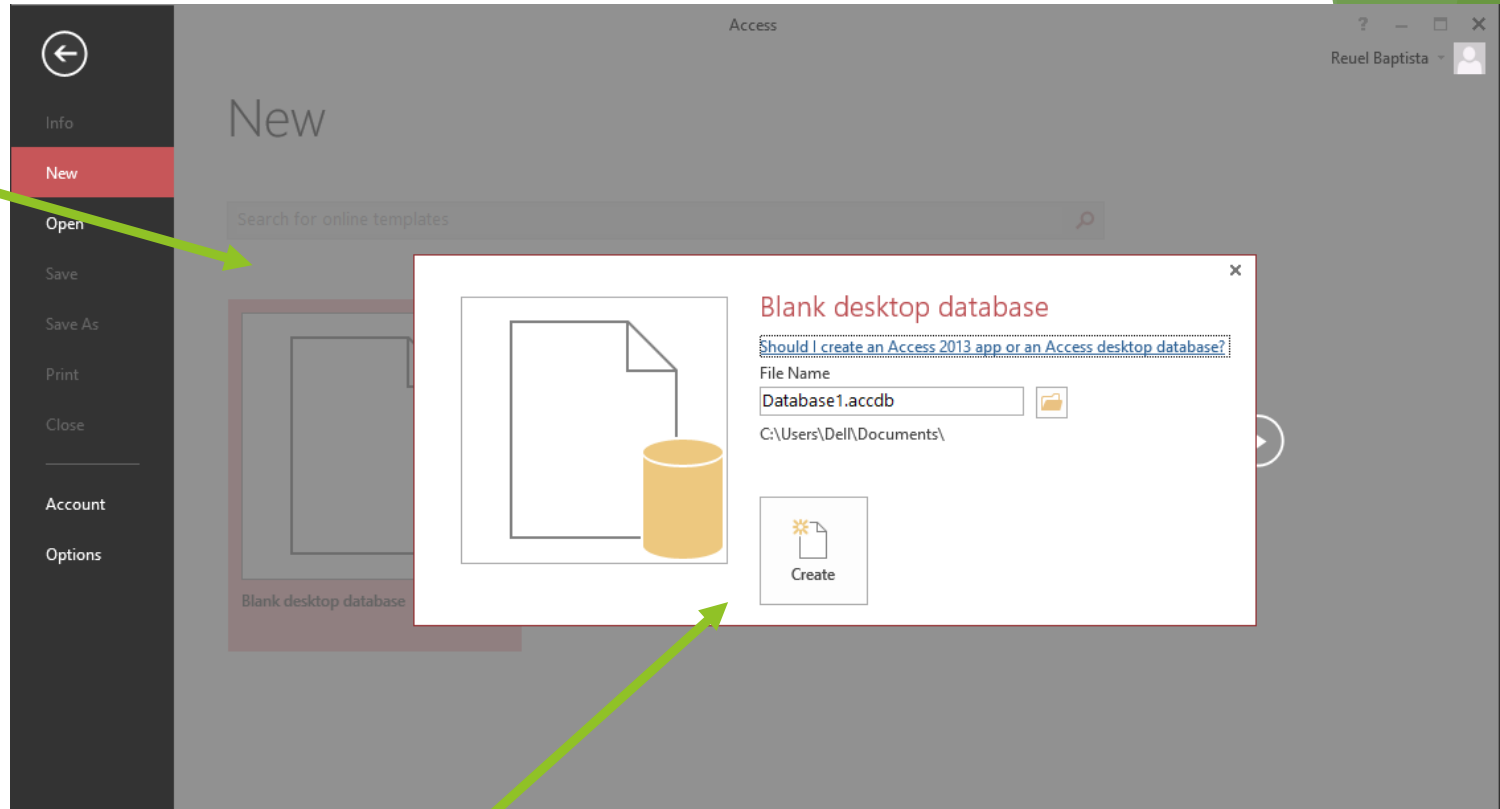
Database Management System (DBMS)

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



Creating a database

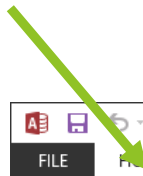
Templates



Creating a new database

Creating a table

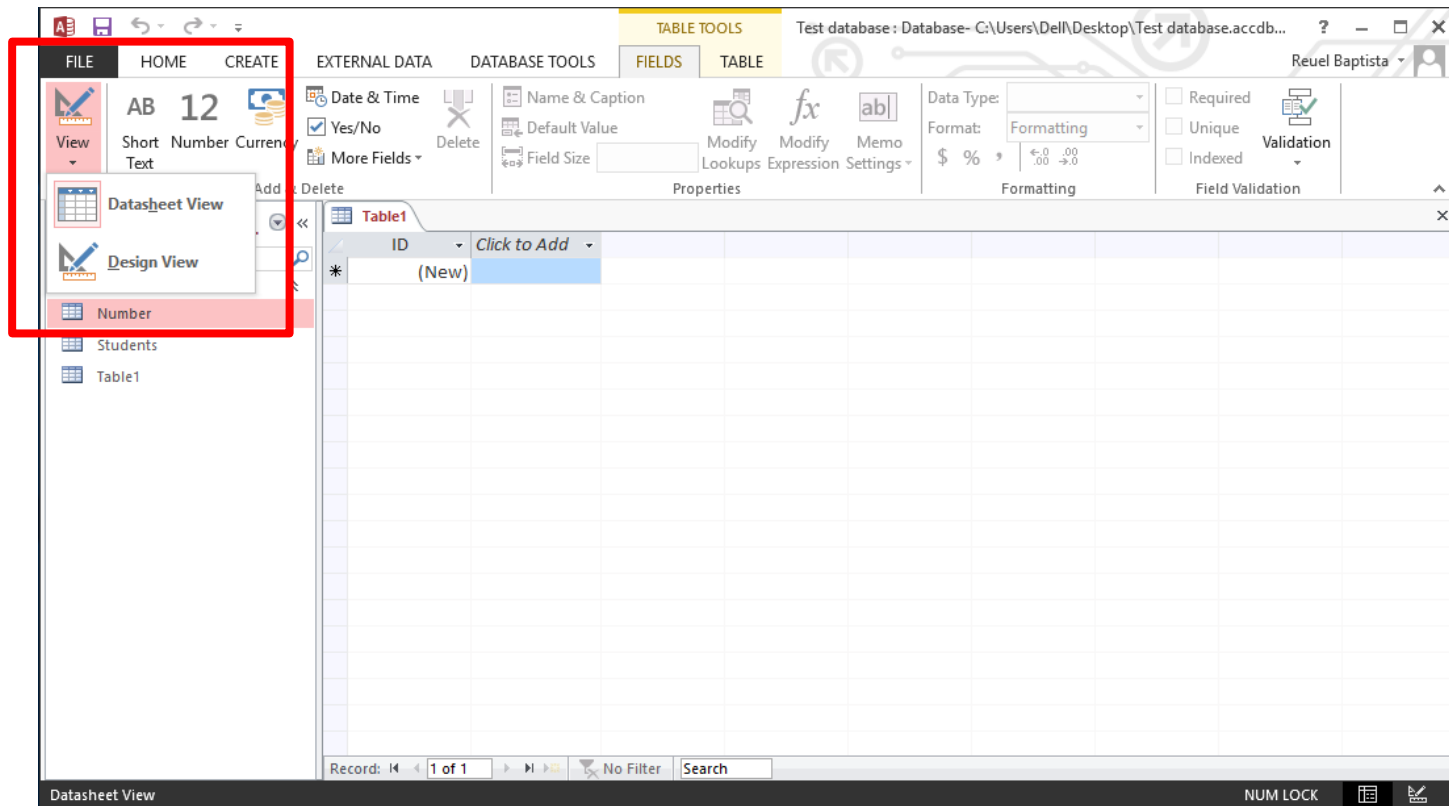
New table
button



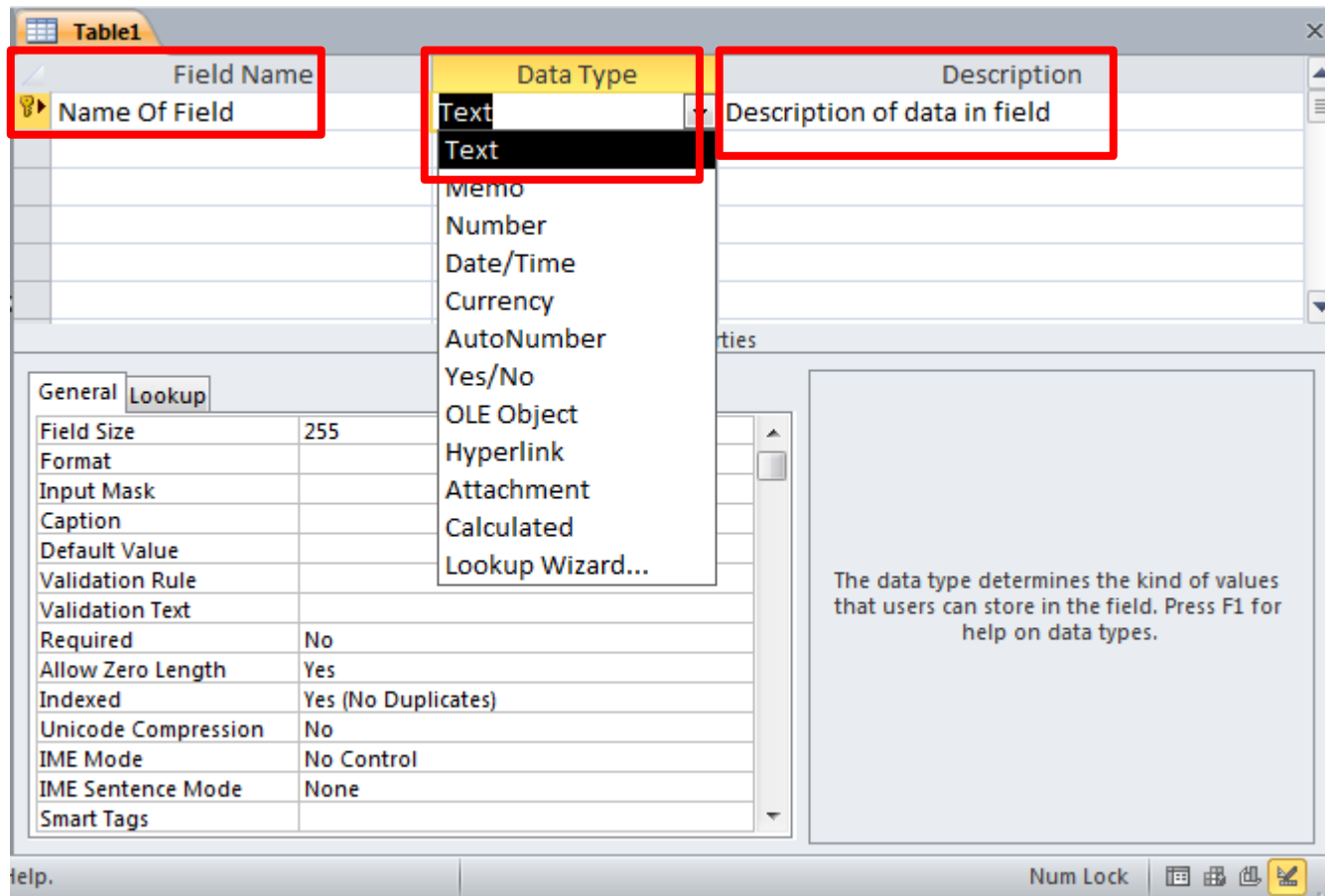
The screenshot shows the Microsoft Access interface. The ribbon is set to the 'CREATE' tab, with the 'Table' button highlighted. The ribbon also includes 'Table Design', 'Table Wizard', 'Form Wizard', 'Report Wizard', and 'Macro'. The 'All Access Objects' pane on the left shows 'Table1' selected. The main window displays a table with one column 'ID' and one row '(New)'. The status bar at the bottom indicates 'Record: 1 of 1' and 'No Filter'.

Creating a table

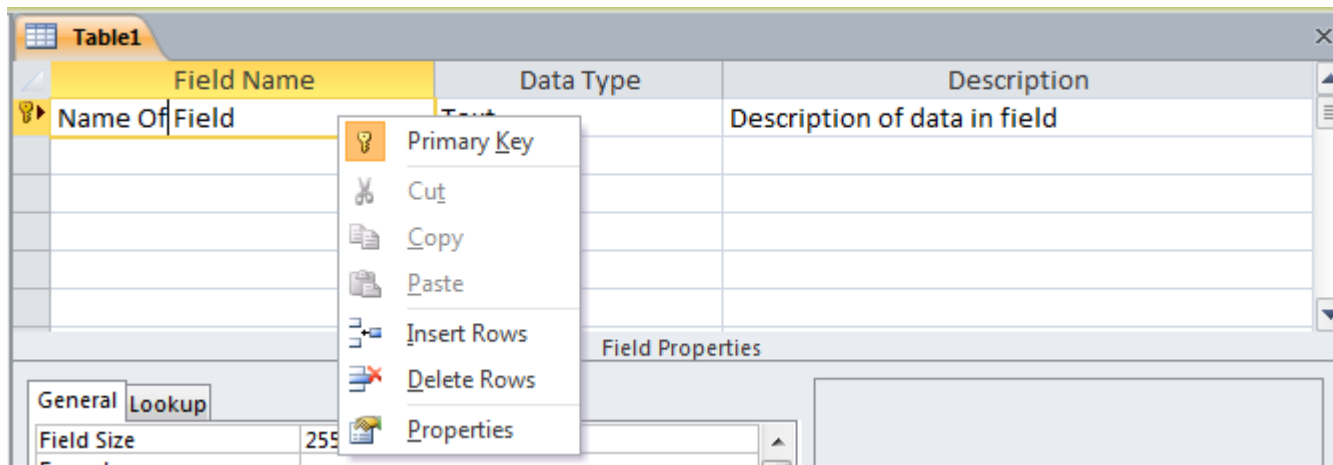
- ▶ Design view: create/view the fields in the table
- ▶ Datasheet view: create/view data in the table



Design view

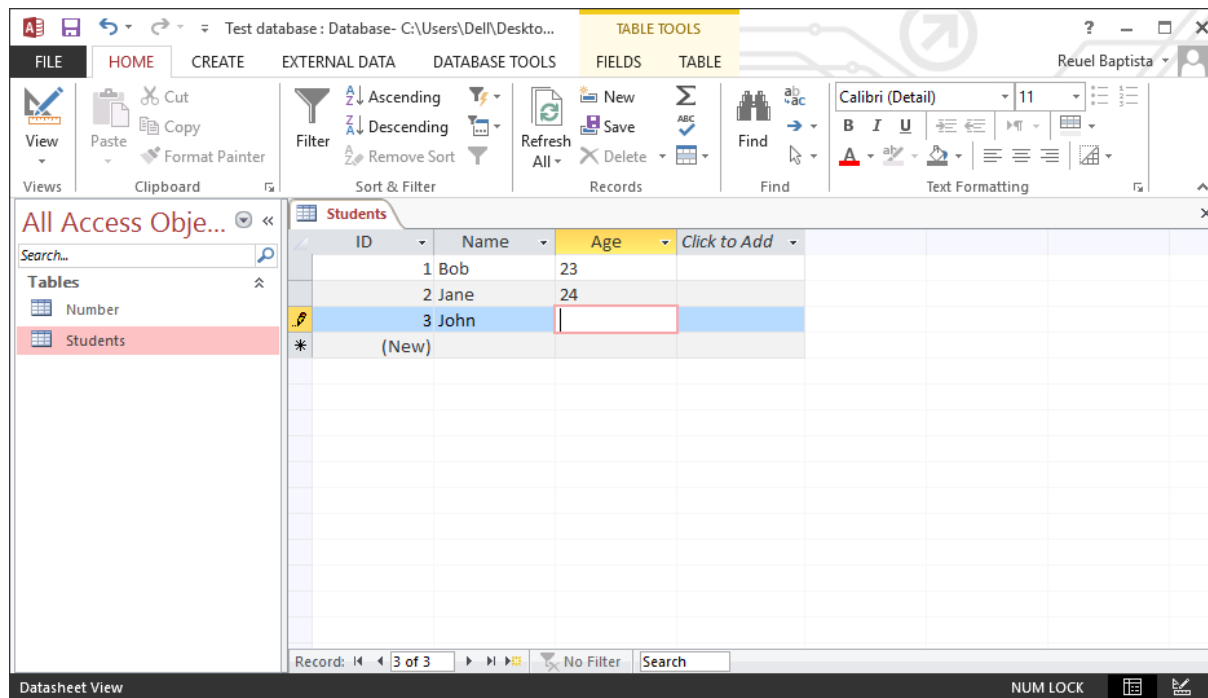


Design view



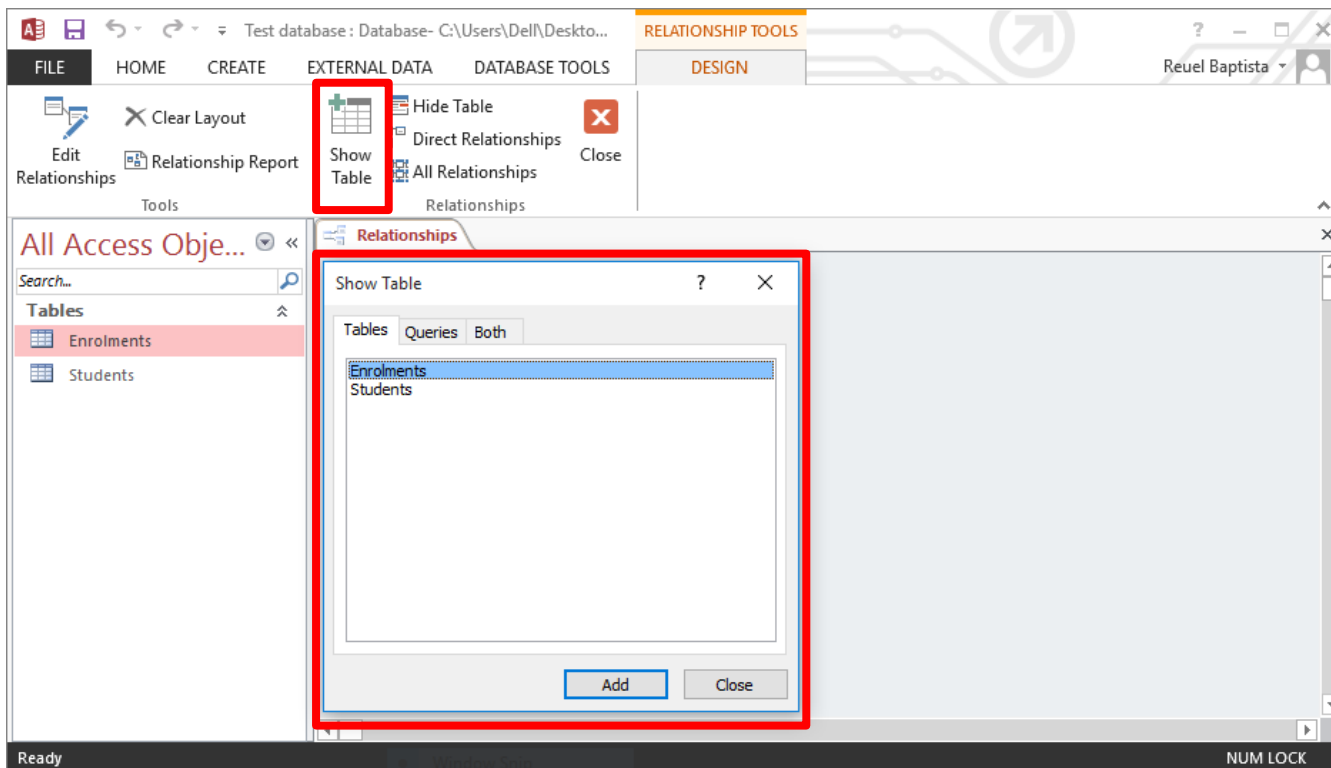
Datasheet view

- ▶ 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

The screenshot displays the Microsoft Access interface in the 'RELATIONSHIP TOOLS' ribbon, specifically the 'DESIGN' tab. The main workspace shows two tables: 'Students' and 'Enrolments'. The 'Students' table has fields 'ID', 'Name', and 'Age'. The 'Enrolments' table has fields 'StudentID', 'Code', and 'Date enrolled'. A green arrow points from the 'ID' field in the 'Students' table to the 'StudentID' field in the 'Enrolments' table, indicating a relationship. On the left, the 'All Access Objects' pane shows 'Tables' with 'Enrolments' and 'Students' listed. The 'Edit Relationships' dialog box is open on the right, showing the relationship between 'Students' and 'Enrolments' with 'ID' and 'StudentID' as the linked fields. The dialog includes options for 'Enforce Referential Integrity', 'Cascade Update Related Fields', and 'Cascade Delete Related Records', and shows the 'Relationship Type' as 'One-To-Many'. Buttons for 'Create', 'Cancel', 'Join Type..', and 'Create New..' are visible.

Test database : Database- C:\Users\DelI\Desкто...
RELATIONSHIP TOOLS
FILE HOME CREATE EXTERNAL DATA DATABASE TOOLS DESIGN
Edit Relationships Relationship Report Show Table Hide Table Direct Relationships All Relationships Close
Tools Relationships
All Access Objects
Search...
Tables
Enrolments
Students
Students
ID
Name
Age
Enrolments
StudentID
Code
Date enrolled
Edit Relationships
Table/Query: Students
Related Table/Query: Enrolments
ID StudentID
Enforce Referential Integrity
Cascade Update Related Fields
Cascade Delete Related Records
Relationship Type: One-To-Many
Create
Cancel
Join Type..
Create New..
Ready NUM LOCK

Creating relationships

The screenshot displays the Microsoft Access interface for creating a relationship between two tables. The 'Relationships' view shows the 'Students' table with fields 'ID', 'Name', and 'Age', and the 'Enrolments' table with fields 'StudentID', 'Code', and 'Date enrolled'. The 'Edit Relationships' dialog box is open, showing the relationship between 'Students' and 'Enrolments'. The 'ID' field in 'Students' is linked to the 'StudentID' field in 'Enrolments'. The 'Enforce Referential Integrity' checkbox is checked, and the 'Relationship Type' is set to 'One-To-Many'. The 'Create' button is highlighted with a red box.

Microsoft Access - Test database: Database- C:\Users\Deff\Desktop...
RELATIONSHIP TOOLS
DESIGN
Reuel Baptista

FILE HOME CREATE EXTERNAL DATA DATABASE TOOLS

Clear Layout Hide Table Direct Relationships Close
Show Table All Relationships

Tools Relationships

All Access Objects
Search...
Tables
Enrolments
Students

Relationships

Students
ID
Name
Age

Enrolments
StudentID
Code
Date enrolled

Edit Relationships

Table/Query: Students Related Table/Query: Enrolments

ID StudentID

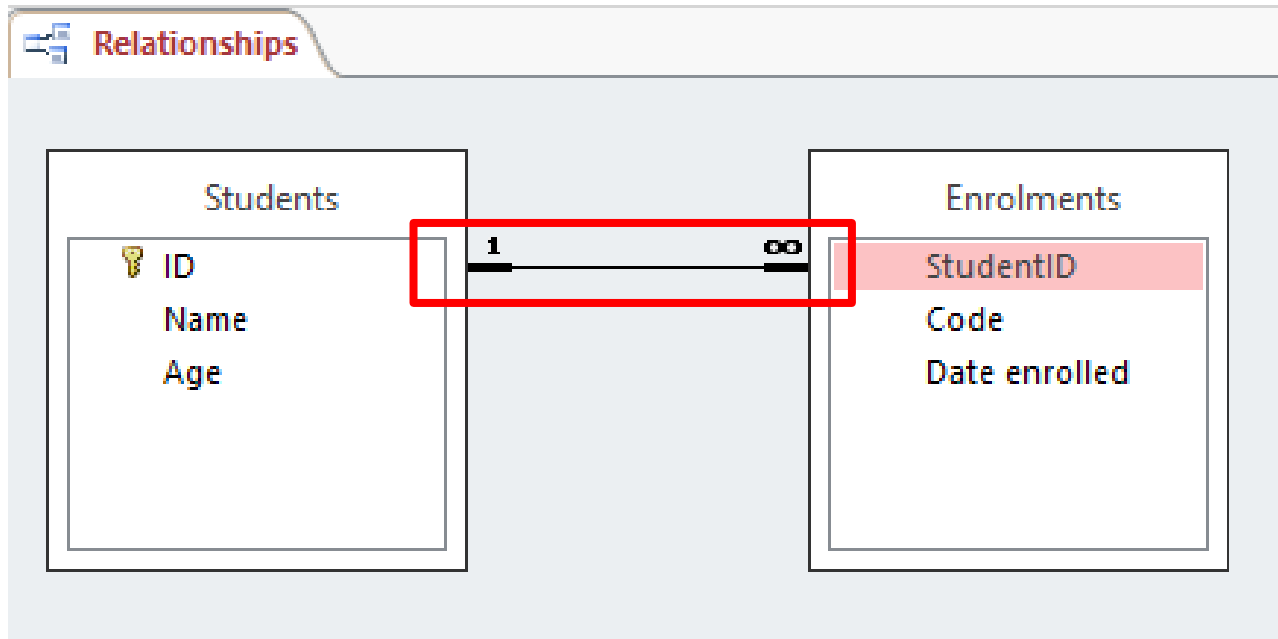
Enforce Referential Integrity
 Cascade Update Related Fields
 Cascade Delete Related Records

Relationship Type: One-To-Many

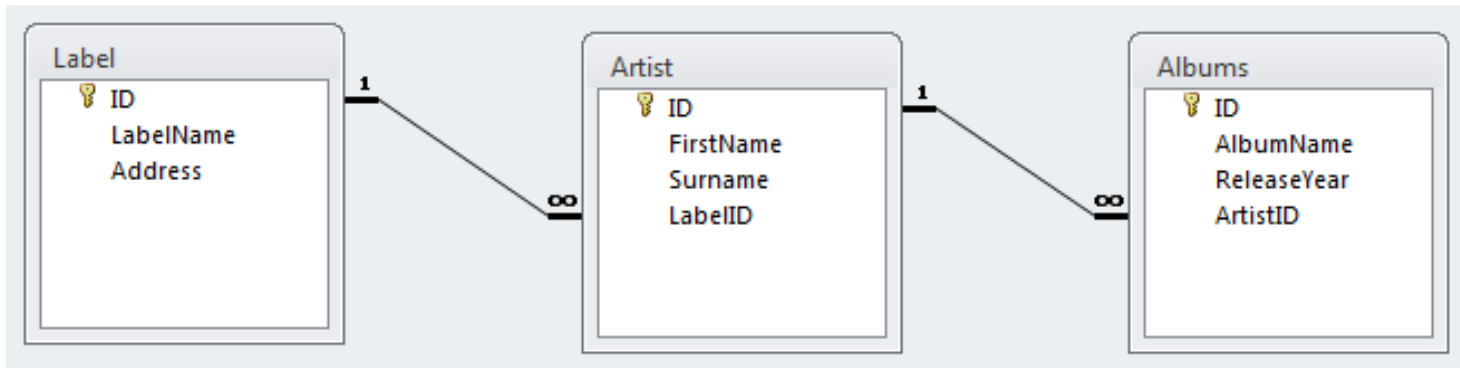
Create
Cancel
Join Type..
Create New..

Ready Window Snip NUM LOCK

Creating relationships

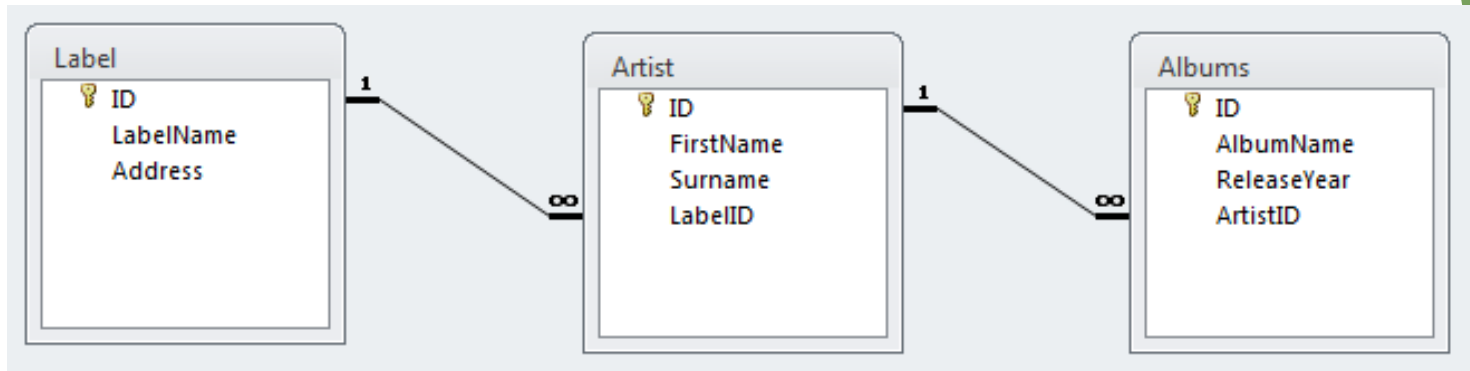


Exercises



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 enroll
	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

The screenshot shows the Microsoft Access interface with two tables: 'Students' and 'Enrolments'. The 'Enrolments' table is active, displaying the following data:

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

An error dialog box titled 'Microsoft Access' is displayed over the table. It contains a yellow warning triangle icon and the message: 'You cannot add or change a record because a related record is required in table 'Students''. The dialog box has 'OK' and 'Help' buttons.

- ▶ 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