CSC343 Worksheet Solution

June 10, 2020

1. **Exercise 2.2.1:** In fig 2.6 are instances of two relations that might constitute part of a banking exercise. Indicate the following

acctNo	*type*	*balance*
12345	savings	12000
23456	checking	1000
34567	savings	25

The relation Accounts

firstName	*lastName*	*idNo*	*account*
Robbie	Banks	901-222	12345
Lena	Hand	805-333	12345
Lena	Hand	805-333	23456

The relation Customers

Figure 2.6: Two relations of a banking database

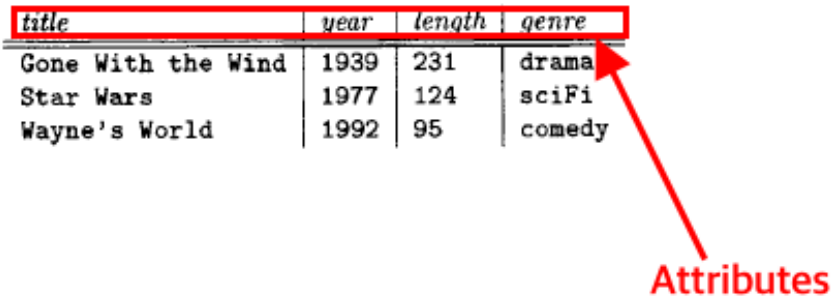
a) The attributes of each relation

Answer:

- Accounts: *acctNo*, *type*, *balance*
- Customers: *firstName*, *lastName*, *idNo*, *account*

Notes:

- **Attributes:** are the columns of a relation



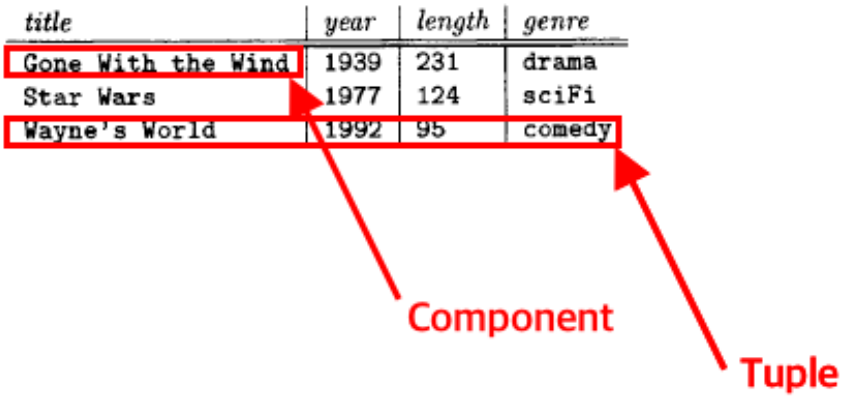
b) The tuples of each relation

Answer:

- Accounts
  1. 12345, savings, 12000
  2. 23456, checkings, 1000
  3. 34567, savings, 25
- Customers
  1. Robbie, Banks, 901-222, 12345
  2. Lena, Hand, 805-333, 12345
  3. Lena, Hand, 805-333, 23456

Notes:

- **Tuple:** the rows of a relation, other than the header row containing the attribute names, are called tuples.
- A tuple has one **component** for each attribute



c) The components of one tuple from each relation

Answer:

1. Accounts
   - 12345
2. Customers
   - Robbie

d) The database schema

Answer:

1. Accounts(acctNo, type, balance)
2. Customers(firstName, lastName, idNo, account)

Notes:

- **Schema:** Is the name of relation and the set of attributes for a relation
  - e.g. Movies(title, year, length, genre)

e) A suitable domain for each attribute

Answer:

1. Accounts (acctNo: integer, type: string, balance: integer)
2. Customers (firstName: string, lastName: string, idNo: String, account: integer)

Notes:

- **Domain:** Is the elementary type of attributes in a relation Schema
  - e.g. Movies(title: string, year: integer, length: integer, genre: string)

f) Another equivalent way to present each relation

Answer:

- Account

type	acctNo	balance
savings	12345	12000
savings	34567	25
checkings	23456	1000

- Customers

account	idNo	lastName	firstName
12345	901-222	Banks	Robbie
23456	805-333	Hand	Lena
12345	805-333	Hand	Lena

Notes:

- Attributes along with its values in re-ordered form has the same relation as the original
- Tuples presented in different order has the same relation as the original

title	*year*	*length*	*genre*
Gone With the Wind	1939	231	drama
Star Wars	1977	124	sciFi
Wayne's World	1992	95	comedy

Figure 2.3: The relation Movies

(is the same)
=

year	*genre*	*title*	*length*
1977	sciFi	Star Wars	124
1992	comedy	Wayne's World	95
1939	drama	Gone With the Wind	231

Figure 2.4: Another presentation of the relation Movies

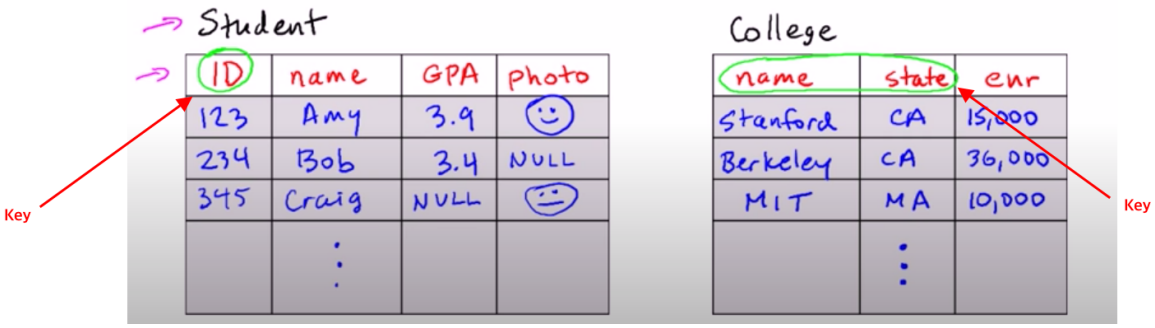
2. **Exercise 2.2.2:** In section 2.2.7 we suggested that there are many exmaples of attributes that are created for the purpose of serving as keys of relations. Give some additional examples.

Answer:

- Account
  - accoutNo
- Customers
  - firstName
  - lastName
  - idNo
  - account

Notes:

- Attribute whose value is unique in each tuple or set of attributes whose combined values are unique



3. **Exericse 2.3.1** In this exercise we introduce one of our running examples of a relational database schema. The database schema consists of four relations, whose schemas are:

```
Product(maker, model, type)
PC(model, speed, ram, hd, price)
Laptop(model, speed, ram, hd, screen, price)
Printer(model, color, type, price)
```

The **Product** relation gives the manufacturer, model number and type (PC¡ laptop, or printer) of various products. We assume for convenience that model numbers are unique over all manufactuers and product types; that assumption is not relaistic, and a real database would include a code for the manufacturer as part of the model number. The PC relation gives for each model number that is a PC the speed (of the processor, in gigahertz), the amount of RAM (in megabytes), the size of the hard disk (in gigabytes), and the price. The **Laptop** relation is similar, except that the screen size (in inches) is also included. The **Printer** relation records for each printer model wether the printer produces color output (true, if so), the process type (laser or ink-jet, typically), and the price.

Write the following declarations:

a) A suitable schema for relation **Product**

b) A suitable schema for relation **Laptop**

c) A suitable schema for relation **Printer**

d) An alteration to your **Printer** schema from (d) to delete the attribute **color**

e) An alternation to your **Laptop** schema from (c) to add the attribute **od** (optical-disk, e.g. cd or dvd). Let the default value for this attribute be '**none**' if the laptop does not have an optical disk.

4. **Exericse 2.3.2** This exercise introduces another running example, concerning World War II capital ships. It involves the following relations:

```
Classes(class, type, country, numGuns, bore, displacement)
Ships(name, class, launched)
Battles(name, date)
Outcomes(ship, battle, result)
```

Ships are bult in 'classes' from the same design, and the class is usually named for the first ship of that class. The relation **Classes** records the name of the class, the type ('bb' for battleship or 'bc' for battlecruiser), the country that built the ship, the number of main guns, the bore (diameter of the gun barrel, in inches) of the main guns, and the displacement (weight, in tones). Relation **Ships** records the name of the ship, the name of its class, and the year in which the ship was launched. Relation **Battles** gives the name and date battles involving these ships, and relation **Outcome** gives the result (sunk, damaged or ok) for each ship in each battle.

Write the following declarations

a) A suitable schema for relation **Classes**

b) A suitable schema for relation **Ships**

c) A suitable schema for relation **Battles**

d) A suitable schema for relation **Outcomes**

e) An alteration to your **Classes** relation from (a) to delete the attribute **bore**

f) An alteration to your **Ships** relation from (b) to include attribute **yard** giving the shipard where the ship was built.