

CSC343 Worksheet 3

June 11, 2020

1. **Exercise 6.1.1:** If a query has a SELECT clause

```
1  SELECT A B
2
```

how do we know whether A and B are two different attributes or B is an alias of A ?

2. **Exercise 6.1.2:** Write the following queries, based on our running movie database example in SQL

```
1  Movies(title, year, length, genre, studioName, producerC\#)
2  StarsIn(movieTitle, movieYear, starName)
3  MovieStar(name, address, gender, birthdate)
4  MovieExec(name, address, cert\#, netWorth)
5  Studio(name, address, presC\#)
6
```

- a) Find the address of MGM studios.
 - b) Find sandra Bullock's birthdate
 - c) Find all the stars that appeared either in a movie made in 1980 or a movie with "Love" in the title
 - d) Find all executives worth at least \$10,000,000
 - e) Find all the stars who either are male or live in malibu (have string *Malibu* as a part of their address)
3. **Exercise 6.1.3:** Write the following queries in SQL. They refer to the database schema of Exercise 2.4.1:

```
1  Product(maker, model type)
2  PC(model, speed, ram, hd, price)
3  Laptop(model, speed, ram, hd, screen, price)
4  Printer(model, color, type, price)
5
```

Show the result of your queries using the data from Exercise 2.4.1

- a) Find the model number, speed and hard-disk size for all PC's whose price is under \$1000
 - b) Do the same as (a), but rename the **speed** column **gigahertz** and the **hd** column **gigabytes**.
 - c) Find the manufacturerers of printers
 - d) Find the model number, memory size, and screen size for laptops costing more than \$1500.
 - e) Find all the tuples in the **Printer** relation for color printers. Remember that **color** is a boolean-valued attribute.
 - f) Find the model nuber and hard-disk size for those PC's that have a speed of 3.2 and a price less than \$2000.
4. **Exercise 6.1.4:** Write the following queries based on the database schema of Exercise 2.4.3:

```

1  Classes(class, type, country, numGuns, bore, displacement)
2  Ships(name, class, launched)
3  Battles(name, date)
4  Outcomes(ship, battle, result)
5

```

and show the result of your query on the data of Exercise 2.4.3

- a) Find the class name and country for all classes with at least 10 guns.
 - b) Find the names of all ships launched prior to 1918, but call the resulting column **shipName**
 - c) Find the names of ships sunk in battle and the name of the battle in which they are sunk
 - d) Find all ships that have the same name as their class
 - e) Find the name of all ships that begin with the letter "R"
 - f) Find the names of all ships whose name consists of three or more words (e.g King George V)
5. **Exercise 6.1.5:** Let a and b be integer-valued attributes that may be *NULL* in some tuples. For each of the following conditions (as may appear in a WHERE clause), describe exactly the set of (a, b) tuples that satisfy the condition, including the case where a and/or b is *NULL*.

- a) $a = 10$ OR $b = 20$
- b) $a = 10$ AND $b = 20$
- c) $a < 10$ OR $a \geq 10$
- d) $a = b$
- e) $a \leq b$

6.

7. **Exercise 6.1.5:** In Example 6.10 we discussed the query

```

1  SELECT *
2  FROM Movies
3  WHERE length <= 120 OR length > 120;
4

```

which behaves unintuitively when the **length** of a movie is *NULL*. Find a simpler, equivalent query, one with a single condition in the **WHERE** clause (no **AND** or **OR** of conditions)

8. **Exercise 6.2.1:** Using the database schema of our running movie example

```

1  Movies(title, year, length, genre, studioName, producerC#)
2  StarsIn(movieTitle, movieYear, starName)
3  MovieStar(name, address, gender, birthdate)
4  MovieExec(name, address, cert\#, netWorth)
5  Studio(name, address, presC#)
6

```

write the following queries in SQL.

- a) Who were the make stars in *Titanic*
- b) Which stars appeared in movie procuded by MGM in 1995
- c) Who is the president of MGM studios?
- d) Which movies are longer than *Gone With the Wind*?
- e) Which executives are worh more than Merv Griffin?

9. **Exercise 6.2.2:** Write the following queries, based on the database schema

```

1  Product(maker, model, type)
2  PC(model, speed, ram, hd, price)
3  Laptop(model, speed, ram, hd, screen, price)
4  Printer(model, color, type, price)
5

```

of Exercise 2.4.1 and evaluate your queries using the data of that exercise.

- a) Give the manufacturer and speed of laptops with a hard disk of at least thirty gigabytes.
- b) Find the model number and price of all products (of any type) made by manufacturer *B*.
- c) Find those manufacturers that sell **Laptops**, but not **PC**'s.
- d) Find those hard-disk sizes that occur in two or more **PC**'s.
- e) Find those pairs of **PC** models that have both the same speed and RAM. A pair should be listed only once; e.g. list (i,j) but not (j, i).

10. **Exercise 6.3.1:** Write the following queries, based on the database schema

```

1  Product(maker, model, type)
2  PC(model, speed, ram, hd, price)
3  Laptop(model, speed, ram, hd, screen, price)
4  Printer(model, color, type, price)
5

```

of Exercise 2.4.1. You should use at least one subquery in each of your answers and write each query in two significantly different ways (e.g. using different sets of the operators **EXIST**, **IN**, **ALL**, and **ANY**)

- a) Find the makers of **PC**'s with a speed of at least 3.0.
- b) Find the printers with the highest price
- c) Find the laptops whose speed is slower than that of any **PC**
- d) Find the model number of the item (PC, laptop, or printer) with the highest price
- e) Find the maker of the color printer with the lowest price

11. **Exercise 6.3.2:** Write the following queries, based on the database schema

```

1  Classes(class, type, country, numGuns, bore, displacement)
2  Ships(name, class, launched)
3  Battles(name, date)
4  Outcomes(ship, battle, result)
5

```

of Exercise 2.4.3. You should use at least one subquery in each of your answers and write each query in two significantly different ways (e.g. using different set of the operators **EXIST**, **IN**, **ALL**, and **ANY**)

- a) Find the countries whose ships had the largest number of guns
- b) Find the classes of ships at least one of which was sunk in a battle
- c) Find the names of the ships with a 16-inch bore
- d) Find the battles in which ships of the Kongo class participated

12. **Exercise 6.3.4** Consider expression $\pi_L(R_1 \bowtie R_2 \bowtie \cdots \bowtie R_n)$ of relational algebra, where L is a list of attributes all of which belong to R_1 . Show that this expression can be written in SQL using subqueries only. More precisely, write an equivalent SQL expression where no **FROM** clause has more than one relation in its list.
13. **Exercise 6.3.5** Write the following queries without using the intersection or difference operators

- a) The intersection query of Fig. 6.5
- b) The difference query of Example 6.17

14. **Exercise 6.3.7** For these relations from our running movie database schema

```

1 StarsIn(movieTitle, movieYear, starName)
2 MovieStar(name, address, gender, birthdate)
3 MovieExec(name, address.cert\#, netWorth)
4 Studio(name, address, pressC\#)
5

```

describe the tuples that would appear in the following SQL expressions:

- a) Studio **CROSS JOIN** MovieExec
- b) StarsIn **NATURAL FULL OUTER JOIN** MovieStar
- c) StarsIn **FULL OUTER JOIN** MovieStar **ON** name = starName;

15. **Exercise 6.3.8** Using the database schema

```

1 Product(maker, model, type)
2 PC(model, speed, ram, hd, price)
3 Laptop(model, speed, ram, hd, screen, price)
4 Printer(model, color, type, price)
5

```

write a SQL query that will produce information about all products - **PC**'s laptops, and printers - including their manufacturer if available, and whatever information about that product is relevant (i.e. found in the relation for that type of product)

16. **Exercise 6.3.9** Using the two relations

```

1 Classes(class, type, country, numGuns, bore, displacement)
2 Ships(name, class, launched)
3

```

from our database schema Exercise 2.4.3 write a SQL query that will produce all available information about ships, including that information available in the **Classes** relation. You need not produce information about classes if there are no ships of that class mentioned in **Ships**.

17. **Exercise 6.3.10** Repeat Exercise 6.3.9 but also include in the result, for any class C that is not mentioned in **Ships**, information about the ship that has the same name C as its class. You may assume that there is a ship with the class name, even if it doesn't appear in **Ships**.

Reference

- 1) Stanford: CS145 - Introduction to Databases, [link](#)