

IT252 Assignment 4

NAME: SUYASH SATISH CHINTAWAR

ROLL NO.: 191IT109

TOPIC: SQL QUERIES

OUTPUT SCREENSHOTS

QUESTION 1(B):

1. Compute the average price of all products with manufacturer code equal to 2.

```
mysql> select avg(price) as average from Products where Manufacturer=2;
+-----+
| average |
+-----+
| 150.0000 |
+-----+
1 row in set (0.00 sec)
```

2. Compute the number of products with a price larger than or equal to \$180.

```
mysql> select count(*) as count from Products where Price>=180;
+-----+
| count |
+-----+
|      5 |
+-----+
1 row in set (0.00 sec)
```

3. Select the name and price of all products with a price larger than or equal to \$180, and sort first by price (in descending order), and then by name (in ascending order).

```
mysql> select name,price from Products where price>=180 order by price desc,name asc;
+-----+-----+
| name      | price |
+-----+-----+
| Printer   | 270   |
| Hard drive | 240   |
| Monitor   | 240   |
| DVD burner | 180   |
| DVD drive | 180   |
+-----+-----+
5 rows in set (0.00 sec)
```

4. Select all the data from the products, including all the data for each product's manufacturer.

```
mysql> select p.Code as p_code,p.Name as p_name,p.Price as p_price,p.Manufacturer as p_manufacturer,m.Name as m_name from Products as p
-> inner join Manufacturers as m
-> on p.Manufacturer=m.Code order by p.Code;
+-----+-----+-----+-----+-----+
| p_code | p_name      | p_price | p_manufacturer | m_name      |
+-----+-----+-----+-----+-----+
| 1      | Hard drive   | 240     | 5              | Fujitsu     |
| 2      | Memory       | 120     | 6              | Winchester  |
| 3      | ZIP drive    | 150     | 4              | Iomega      |
| 4      | Floppy disk  | 5        | 6              | Winchester  |
| 5      | Monitor      | 240     | 1              | Sony        |
| 6      | DVD drive    | 180     | 2              | Creative Labs |
| 7      | CD drive     | 90      | 2              | Creative Labs |
| 8      | Printer      | 270     | 3              | Hewlett-Packard |
| 9      | Toner cartridge | 66      | 3              | Hewlett-Packard |
| 10     | DVD burner   | 180     | 2              | Creative Labs |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

5. Select the average price of each manufacturer's products, showing only the manufacturer's code.

```
mysql> select Manufacturer,avg(price) from Products group by Manufacturer;
+-----+-----+
| Manufacturer | avg(price) |
+-----+-----+
| 1           | 240.0000   |
| 2           | 150.0000   |
| 3           | 168.0000   |
| 4           | 150.0000   |
| 5           | 240.0000   |
| 6           | 62.5000    |
+-----+-----+
6 rows in set (0.00 sec)
```

6. Select the product name, price, and manufacturer name of all the products.

```
mysql> select p.Name as p_name,p.Price as p_price,m.name as m_name from Products as p
-> inner join Manufacturers as m
-> on p.Manufacturer=m.Code order by p.name;
+-----+-----+-----+
| p_name | p_price | m_name |
+-----+-----+-----+
| CD drive | 90 | Creative Labs |
| DVD burner | 180 | Creative Labs |
| DVD drive | 180 | Creative Labs |
| Floppy disk | 5 | Winchester |
| Hard drive | 240 | Fujitsu |
| Memory | 120 | Winchester |
| Monitor | 240 | Sony |
| Printer | 270 | Hewlett-Packard |
| Toner cartridge | 66 | Hewlett-Packard |
| ZIP drive | 150 | Iomega |
+-----+-----+-----+
10 rows in set (0.00 sec)
```

7. Select the names of manufacturer whose products have an average price larger than or equal to \$150.

```
mysql> select m.Name from Products as p
-> inner join Manufacturers as m
-> on p.Manufacturer=m.Code group by m.name having avg(price)>=150;
+-----+
| Name |
+-----+
| Sony |
| Creative Labs |
| Hewlett-Packard |
| Iomega |
| Fujitsu |
+-----+
5 rows in set (0.01 sec)
```

8. Select the name and price of the cheapest product.

```
mysql> select name,price from Products where price=(select min(price) from Products);
+-----+-----+
| name      | price |
+-----+-----+
| Floppy disk |     5 |
+-----+-----+
1 row in set (0.00 sec)
```

9. Select the name of each manufacturer along with the name and price of its most expensive product.

```
mysql> select m.Name as m_name,p.name as p_name ,p.price from Products as p
-> inner join Manufacturers as m on p.Manufacturer=m.Code
-> where p.code=any(select p.code from Products as p inner join Manufacturers as m on p.Manufacturer=m.Code
-> where p.price=any(select max(p.price) from Products as p inner join Manufacturers as m on p.Manufacturer=m.Code group by p.Manufacturer));
+-----+-----+-----+
| m_name      | p_name      | price |
+-----+-----+-----+
| Sony         | Monitor     |    240 |
| Creative Labs | DVD drive   |    180 |
| Creative Labs | DVD burner  |    180 |
| Hewlett-Packard | Printer    |    270 |
| Iomega       | ZIP drive   |    150 |
| Fujitsu      | Hard drive  |    240 |
| Winchester   | Memory      |    120 |
+-----+-----+-----+
7 rows in set (0.00 sec)
```

10. Apply a 10% discount to all products with a price larger than or equal to \$120.

```
mysql> update Products set price=0.9*price where price>=120;
Query OK, 7 rows affected (0.01 sec)
Rows matched: 7  Changed: 7  Warnings: 0

mysql> select * from Products;
+-----+-----+-----+-----+
| Code | Name           | Price | Manufacturer |
+-----+-----+-----+-----+
| 1    | Hard drive     |    216 | 5            |
| 2    | Memory         |    108 | 6            |
| 3    | ZIP drive      |    135 | 4            |
| 4    | Floppy disk     |     5  | 6            |
| 5    | Monitor        |    216 | 1            |
| 6    | DVD drive      |    162 | 2            |
| 7    | CD drive       |     90 | 2            |
| 8    | Printer        |    243 | 3            |
| 9    | Toner cartridge |     66 | 3            |
| 10   | DVD burner     |    162 | 2            |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

QUESTION 2(B):

1. Select the number of employees in each department (you only need to show the department code and the number of employees).

```
mysql> select Department,count(*) as count from Employees group by Department;
+-----+-----+
| Department | count |
+-----+-----+
|          14 |      5 |
|          37 |      2 |
|          59 |      3 |
|          77 |      2 |
+-----+-----+
4 rows in set (0.00 sec)
```

2. Select the name and last name of each employee, along with the name and budget of the employee's department.

```
mysql> select e.Name as e_name,e.LastName,d.name as d_name,d.budget from Employees as e
-> inner join Departments as d on e.Department=d.Code;
+-----+-----+-----+-----+
| e_name | LastName | d_name | budget |
+-----+-----+-----+-----+
| Michael | Rogers | IT | 65000 | |
| Anand | Manikutty | IT | 65000 |
| Mary- | Anne | Foster | IT | 65000 |
| Elizabeth | Doe | IT | 65000 |
| Kumar | Swamy | IT | 65000 |
| Carol | Smith | Accounting | 15000 |
| Joe | Stevens | Accounting | 15000 |
| John | Doe | Human Resources | 240000 |
| Zacary | Efron | Human Resources | 240000 |
| Eric | Goldsmith | Human Resources | 240000 |
| George | ODonnell | Research | 55000 |
| David | Smith | Research | 55000 |
+-----+-----+-----+-----+
12 rows in set (0.00 sec)
```

3. Select the name and last name of employees working for departments with a budget greater than \$60,000

```
mysql> select e.Name,e.LastName from Employees as e inner join Departments as d on e.Department=d.Code where d.budget>=60000;
+-----+-----+
| Name | LastName |
+-----+-----+
| Michael | Rogers | |
| Anand | Manikutty |
| Mary- | Anne | Foster |
| Elizabeth | Doe |
| Kumar | Swamy |
| John | Doe |
| Zacary | Efron |
| Eric | Goldsmith |
+-----+-----+
8 rows in set (0.00 sec)
```

4. Select the departments with a budget larger than the average budget of all the departments

```
mysql> select * from Departments where budget>(select avg(budget) from Departments);
+-----+-----+-----+
| Code | Name          | Budget |
+-----+-----+-----+
| 59   | Human Resources | 240000 |
+-----+-----+-----+
1 row in set (0.00 sec)
```

5. Select the names of departments with more than two employees.

```
mysql> select name from Departments where code in(select Department from Employees group by Department having count(*)>2);
+-----+
| name |
+-----+
| IT   |
| Human Resources |
+-----+
2 rows in set (0.00 sec)
```

6. Select the name and last name of employees working for departments with second lowest budget

```
mysql> select name,lastname from Employees where department=(select code from Departments
-> where budget=(select budget from Departments order by budget asc limit 1,1));
+-----+-----+
| name | lastname |
+-----+-----+
| George | ODonnell |
| David | Smith   |
+-----+-----+
2 rows in set (0.00 sec)
```

7. Reduce the budget of all departments by 10%

```
mysql> update Departments set budget=0.9*budget;
Query OK, 4 rows affected (0.00 sec)
Rows matched: 4  Changed: 4  Warnings: 0

mysql> select * from Departments;
+-----+-----+-----+
| Code | Name          | Budget |
+-----+-----+-----+
| 14   | IT            | 58500  |
| 37   | Accounting    | 13500  |
| 59   | Human Resources | 216000 |
| 77   | Research      | 49500  |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

8. Reassign all employees from the Research department (code 77) to the IT department (code 14).

```
mysql> update Employees set Department=14 where Department=77;
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> select * from Employees;
+-----+-----+-----+-----+
| SSN      | Name      | LastName | Department |
+-----+-----+-----+-----+
| 123234877 | Michael   | Rogers   | 14         |
| 152934485 | Anand     | Manikutty | 14         |
| 222364883 | Carol     | Smith    | 37         |
| 326587417 | Joe       | Stevens  | 37         |
| 332154719 | Mary-     | Anne     | 14         |
| 332569843 | George    | ODonnell | 14         |
| 546523478 | John      | Doe       | 59         |
| 631231482 | David     | Smith    | 14         |
| 654873219 | Zacary    | Efron     | 59         |
| 745685214 | Eric      | Goldsmith | 59         |
| 845657245 | Elizabeth | Doe       | 14         |
| 845657246 | Kumar     | Swamy     | 14         |
+-----+-----+-----+-----+
12 rows in set (0.00 sec)
```

9. Add a new department called "Quality Assurance", with a budget of \$40,000 and departmental code 11. And Add an employee called "Mary Moore" in that department, with SSN 847-21-9811.

```
mysql> insert into Departments values (11,'Quality Assurance',40000);
Query OK, 1 row affected (0.01 sec)

mysql> select * from Departments;
+-----+-----+-----+
| Code | Name           | Budget |
+-----+-----+-----+
| 11    | Quality Assurance | 40000  |
| 14    | IT              | 58500  |
| 37    | Accounting      | 13500  |
| 59    | Human Resources | 216000 |
| 77    | Research        | 49500  |
+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> insert into Employees values (847219811,'Mary','Moore',11);
Query OK, 1 row affected (0.01 sec)

mysql> select * from Employees;
+-----+-----+-----+-----+
| SSN      | Name      | LastName | Department |
+-----+-----+-----+-----+
| 123234877 | Michael   | Rogers   | 14         |
| 152934485 | Anand     | Manikutty | 14         |
| 222364883 | Carol     | Smith    | 37         |
| 326587417 | Joe       | Stevens  | 37         |
| 332154719 | Mary-     | Anne     | 14         |
| 332569843 | George    | ODonnell | 14         |
| 546523478 | John      | Doe       | 59         |
| 631231482 | David     | Smith    | 14         |
| 654873219 | Zacary    | Efron     | 59         |
| 745685214 | Eric      | Goldsmith | 59         |
| 845657245 | Elizabeth | Doe       | 14         |
| 845657246 | Kumar     | Swamy     | 14         |
| 847219811 | Mary      | Moore     | 11         |
+-----+-----+-----+-----+
13 rows in set (0.00 sec)
```

10. Delete from the table all employees who work in departments with a budget greater than or equal to \$60,000.

```
mysql> delete from Employees where department in (select code from Departments where budget>=60000);
Query OK, 3 rows affected (0.01 sec)

mysql> select * from Employees;
+-----+-----+-----+-----+
| SSN      | Name      | LastName | Department |
+-----+-----+-----+-----+
| 123234877 | Michael   | Rogers   | 14         |
| 152934485 | Anand     | Manikutty | 14         |
| 222364883 | Carol     | Smith    | 37         |
| 326587417 | Joe       | Stevens  | 37         |
| 332154719 | Mary-     |          |            |
Anne | Foster    | 14       |            |
| 332569843 | George    | ODonnell | 14         |
| 631231482 | David     | Smith    | 14         |
| 845657245 | Elizabeth | Doe      | 14         |
| 845657246 | Kumar     | Swamy    | 14         |
| 847219811 | Mary      | Moore    | 11         |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

NOTE: Log file has also been attached in the folder. File name is
“191IT109_ChintawarSuyashSatish_SQLQueries.txt”