DBMS - 1

Lab Report 1

SUBMITTED BY:
SHAHZANEER AHMED

REGISTRATION NUMBER:
SP21-BCS-087

SUBMITTED TO:
Dr. Basit Raza

DATE OF SUBMISSION:
October 13, 2022

// Question 1

List of PC models greater than 1000

$$(\pi \text{ model}(\sigma \text{ model} > 1000(PC)))$$

List of PC models greater than 1000 and HD greater than 250.

$$\pi \text{ model}(\sigma \text{ model} > 1000 \wedge \text{hd} > 250 (PC))$$

List of PC model, ram, hd and Price where model should be greater than 1000 and hd should be greater than 250

$$\pi \text{ model,ram,price,hd}(\sigma \text{ model} > 1000 \wedge \text{hd} > 250 (PC))$$

Find Maker, Model, Type, Speed, Ram and Price of Laptops which are released after the year 2000, have a Price more than 1000 and have a Speed greater than 1.5.

$$\pi \text{ Product.maker,Laptop.model,Product.type,speed,ram,price}(\sigma \text{ price} > 1000 \wedge \text{speed} > 1.5 \wedge \text{Laptop.model} > 2000 (\text{Laptop} \bowtie \text{Product}))> 1.5 \wedge \text{Laptop.model} > 2000 (\text{Laptop} \bowtie \text{Product}))> 1000(PC)$$

List of product maker, PC model, ram, hd and price. Where model should be greater than 1000 and hd should be greater than 250 and maker should not be

'B'.

$$\pi \text{ maker,model,ram,hd,price}(\sigma \text{ model} > 1000 \wedge \text{hd} > 250 \wedge \text{maker} \neq \text{'B'}(\text{Product} \bowtie PC))$$

Find the Speed of Laptops and PCs.

$$(\rho \text{ ALL } ((\pi \text{ PC.speed } (PC) \cup \pi \text{Laptop.speed}(\text{Laptop}))))$$

List the Models of all the Laptops and PCs those have same speed.

$$\rho \text{ AllPCAndLaptopModels } \pi \text{ PC.model}(\sigma \text{ PC.speed} = \text{Laptop.speed } (PC \times \text{Laptop})) \cup \pi \text{ Laptop.model}(\sigma \text{ PC.speed} = \text{Laptop.speed } (PC \times \text{Laptop}))$$

List the Model and Maker of all the Laptops who have Model > 2005.

$$\pi \text{ model, maker } (\sigma \text{ model} > 2005 \text{ (Laptop} \bowtie \text{Product)})$$

List Product Model along with their Product Maker of all the Products with a Speed greater than 1.5.

$$\pi \text{ model, maker } (\sigma \text{ speed} > 1.5 \text{ (Product} \bowtie \text{PC)}) \cup \pi \text{ model, maker } (\sigma \text{ speed} > 1.5 \text{ (Product} \bowtie \text{Laptop))}$$

List Model of all the Products that are released after 2000, have a Speed greater than 1.5, and that belong to Laptop Type.

$$\rho \text{ MODELS}(\pi \text{ model } (\sigma \text{ model} > 2000 \wedge \text{speed} > 1.5 \wedge \text{type} = \text{'laptop'} \text{ (Laptop} \bowtie \text{Product))}$$

// Question 2

Display the last name, job_id, hire_date, and employee_id for each employee, with the employee_id appearing first. Rename HIRE_DATE column as STARTDATE.

```
SELECT employee_id, last_name, job_id, hire_date as "STARTDATE" from
EMPLOYEES;
```

Display the last_name concatenated with the job_id (separated by a comma and space) of all the employees and name the columns as Employee and Title.

```
SELECT last_name ||', '|| job_id "Employee and Title" from EMPLOYEES;
```

Display the structure of the departments table.

```
DESCRIBE departments;
```

Instead of retrieving all 107 rows from employees table for all job_id's, find only distinct job_ids.

```
SELECT DISTINCT job_id from EMPLOYEES;
```

Show first name and last name of all employees after concatenation them. Use space as the separator. Name the resultant column as “Name”

```
SELECT first_name ||' '|| last_name as "Name" from EMPLOYEES;
```

Display first name and the annual salary of employees.

```
SELECT first_name, salary*12 "Annual Salary" from EMPLOYEES;
```

// Question 3

Retrieve the last name and department number for employee number 176.

```
SELECT last_name, department_id FROM EMPLOYEES
where employee_id = 176;
```

Retrieve the last name and salary for any employee whose salary is not in the range of 5,000 to 12,000.

```
SELECT last_name, salary FROM employees
WHERE salary NOT BETWEEN 5000 AND 12000;
```

Retrieve the last name, job ID, and hire date for employees with the last names of Matos or Taylor. Sort the resultant rows in ascending order by the hire date.

```
SELECT last_name, job_id, hire_date FROM employees
WHERE last_name IN ('Matos', 'Taylor')
ORDER BY hire_date;
```

Retrieve the last name and department ID of all employees in departments 20 or 50 in ascending alphabetical order by name.

```
SELECT last_name , department_id FROM employees
Where department_id IN (20, 50)
Order by first_name||last_name ASC ;
```

Retrieve the last name and job title of all employees whose department ID is unknown.

```
SELECT last_name, job_id FROM employees
WHERE department_id IS NULL;
```

Retrieve the last name, salary, and commission of all employees who earn commissions (i.e., the commission_pct is not NULL). Sort the resultant rows in descending order of salary and then commissions. Use the salary column's numeric position in the ORDER BY clause.

```
SELECT last_name, salary, commission_pct FROM employees
where commission_pct is NOT NULL
ORDER BY salary DESC , commission_pct;
```

Write a query that prompts the user for a manager ID and retrieves the employee ID, last name, salary, and department for that manager's employees. The query must also prompt the column based on which the resultant rows will be sorted in ascending order.

```
SELECT employee_id, last_name, salary, department_id from employees
where manager_id = '&managerID'
order by &columnName;
```

Retrieve the last names of all employees who have both an "a" and an "e" somewhere in any order in their last name.

```
SELECT last_name
FROM employees
WHERE last_name LIKE '%a%' AND last_name LIKE '%e%';
```

Retrieve the last name, job, and salary for all employees whose jobs are either those of a sales representative or of a stock clerk, and whose salaries are not equal to 2500, 3500, or 7000.

```
SELECT last_name, job_id, salary
from employees
where job_id IN ('SA_REP', 'ST_CLERK')
```