WEEK-2 ASSIGNMENTS

Khushbu Patra

SKILL:-SQL-ADVANCED CONCEPTS

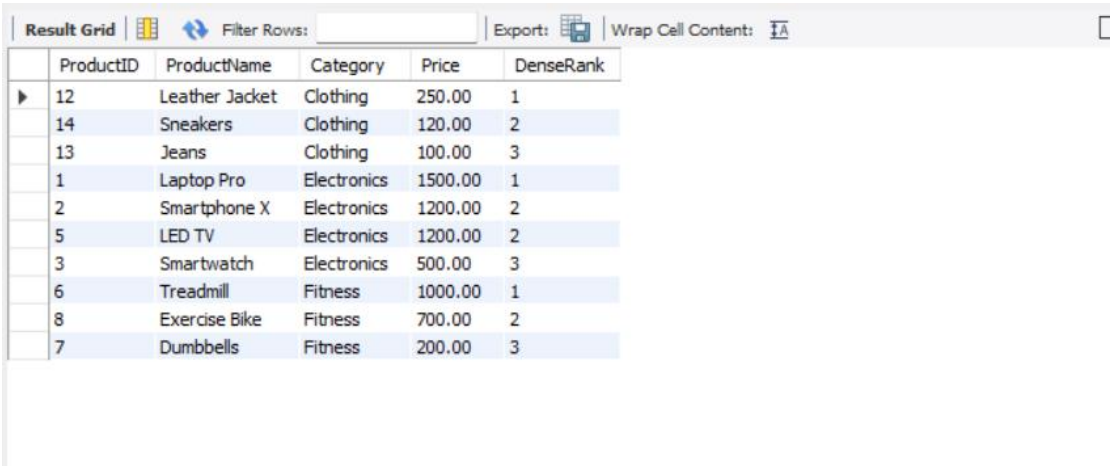
EXERCISE 1:-RANKING AND WINDOW FUNCTIONS:-

SQL CODE:-

```
create database cognizant;
use cognizant;
CREATE TABLE Products (
   ProductID INT,
   ProductName VARCHAR(100),
   Category VARCHAR(100),
   Price DECIMAL(10,2)
);
SELECT
   ProductID,
   ProductName,
   Category,
   Price,
   ROW_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS
RowNum
FROM Products;

SELECT *
FROM (
   SELECT
      ProductID,
      ProductName,
      Category,
      Price,
      ROW_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS
RowNum
   FROM Products
) AS Ranked
WHERE RowNum <= 3;
INSERT INTO Products (ProductID, ProductName, Category, Price) VALUES
(1, 'Laptop Pro',        'Electronics', 1500.00),
(2, 'Smartphone X',      'Electronics', 1200.00),
(3, 'Smartwatch',        'Electronics', 500.00),
(4, 'Bluetooth Speaker',  'Electronics', 300.00),
(5, 'LED TV',            'Electronics', 1200.00),

(6, 'Treadmill',         'Fitness',     1000.00),
(7, 'Dumbbells',         'Fitness',     200.00),
```

```
(8, 'Exercise Bike',     'Fitness',     700.00),
(9, 'Yoga Mat',          'Fitness',     100.00),
(10, 'Resistance Bands',  'Fitness',     100.00),

(11, 'Formal Shirt',     'Clothing',    70.00),
(12, 'Leather Jacket',   'Clothing',    250.00),
(13, 'Jeans',            'Clothing',    100.00),
(14, 'Sneakers',         'Clothing',    120.00),
(15, 'T-shirt',          'Clothing',    50.00);

SELECT *
FROM (
  SELECT
    ProductID,
    ProductName,
    Category,
    Price,
    RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS rnk
  FROM Products
) AS ranked_products
WHERE rnk <= 3;

SELECT *
FROM (
  SELECT
    ProductID,
    ProductName,
    Category,
    Price,
    DENSE_RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS
DenseRank
  FROM Products
) AS Ranked
WHERE DenseRank <= 3;
```

OUTPUT:-



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProductID	ProductName	Category	Price	DenseRank
12	Leather Jacket	Clothing	250.00	1
14	Sneakers	Clothing	120.00	2
13	Jeans	Clothing	100.00	3
1	Laptop Pro	Electronics	1500.00	1
2	Smartphone X	Electronics	1200.00	2
5	LED TV	Electronics	1200.00	2
3	Smartwatch	Electronics	500.00	3
6	Treadmill	Fitness	1000.00	1
8	Exercise Bike	Fitness	700.00	2
7	Dumbbells	Fitness	200.00	3

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProductID	ProductName	Category	Price	rnk
12	Leather Jacket	Clothing	250.00	1
14	Sneakers	Clothing	120.00	2
13	Jeans	Clothing	100.00	3
1	Laptop Pro	Electronics	1500.00	1
2	Smartphone X	Electronics	1200.00	2
5	LED TV	Electronics	1200.00	2
6	Treadmill	Fitness	1000.00	1
8	Exercise Bike	Fitness	700.00	2
7	Dumbbells	Fitness	200.00	3

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProductID	ProductName	Category	Price	RowNum
12	Leather Jacket	Clothing	250.00	1
14	Sneakers	Clothing	120.00	2
13	Jeans	Clothing	100.00	3
1	Laptop Pro	Electronics	1500.00	1
2	Smartphone X	Electronics	1200.00	2
5	LED TV	Electronics	1200.00	3
6	Treadmill	Fitness	1000.00	1
8	Exercise Bike	Fitness	700.00	2
7	Dumbbells	Fitness	200.00	3

Result 3

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ProductID	ProductName	Category	Price	RowNum
12	Leather Jacket	Clothing	250.00	1
14	Sneakers	Clothing	120.00	2
13	Jeans	Clothing	100.00	3
11	Formal Shirt	Clothing	70.00	4
15	T-shirt	Clothing	50.00	5
1	Laptop Pro	Electronics	1500.00	1
2	Smartphone X	Electronics	1200.00	2
5	LED TV	Electronics	1200.00	3
3	Smartwatch	Electronics	500.00	4
4	Bluetooth Speaker	Electronics	300.00	5
6	Treadmill	Fitness	1000.00	1
8	Exercise Bike	Fitness	700.00	2
7	Dumbbells	Fitness	200.00	3
9	Yoga Mat	Fitness	100.00	4

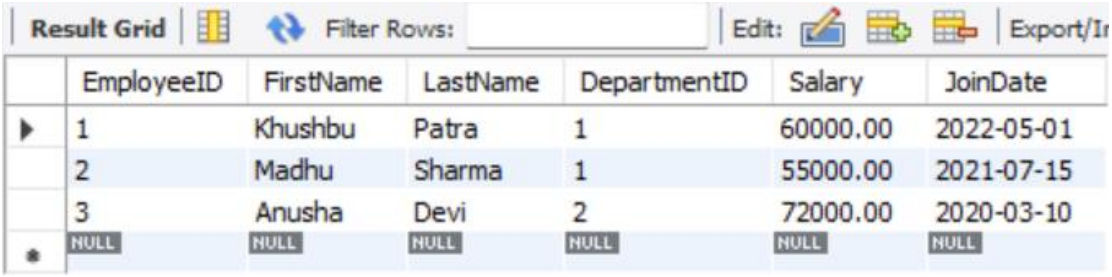
Result 2

Exercise 1: Create a Stored Procedure

SQL CODE:-

```
CREATE TABLE Employees (
   EmployeeID INT AUTO_INCREMENT PRIMARY KEY,
   FirstName VARCHAR(50),
```

```
   LastName VARCHAR(50),
   DepartmentID INT,
   Salary DECIMAL(10,2),
   JoinDate DATE
);

DELIMITER //

CREATE PROCEDURE sp_GetEmployeesByDepartment (
   IN deptId INT
)
BEGIN
   SELECT * FROM Employees
   WHERE DepartmentID = deptId;
END //

DELIMITER ;

INSERT INTO Employees (FirstName, LastName, DepartmentID, Salary, JoinDate)
VALUES
('Khushbu', 'Patra', 1, 60000.00, '2022-05-01'),
('Madhu', 'Sharma', 1, 55000.00, '2021-07-15'),
('Anusha', 'Devi', 2, 72000.00, '2020-03-10');

SELECT * FROM Employees;
```



EmployeeID	FirstName	LastName	DepartmentID	Salary	JoinDate
1	Khushbu	Patra	1	60000.00	2022-05-01
2	Madhu	Sharma	1	55000.00	2021-07-15
3	Anusha	Devi	2	72000.00	2020-03-10
NULL	NULL	NULL	NULL	NULL	NULL

```
CALL sp_GetEmployeesByDepartment(1);
```

EmployeeID	FirstName	LastName	DepartmentID	Salary	JoinDate
1	Khushbu	Patra	1	60000.00	2022-05-01
2	Madhu	Sharma	1	55000.00	2021-07-15

EXERCISE 5: Count of Employees in a Department

SQL CODE:-

```
DELIMITER //
```

```
CREATE PROCEDURE sp_CountEmployeesByDepartment (
   IN deptId INT
)
BEGIN
   SELECT COUNT(*) AS TotalEmployees
   FROM Employees
   WHERE DepartmentID = deptId;
END //

DELIMITER ;

CALL sp_CountEmployeesByDepartment(1);
```


```
CALL sp_CountEmployeesByDepartment(2);
```