# Q1

```sql
-- a. Create the Books table
CREATE TABLE Books (
    BookID NUMBER PRIMARY KEY,
    Title VARCHAR2(200),
    Author VARCHAR2(100)
);

-- b. Insert 5 records into the Books table with less popular titles
INSERT INTO Books VALUES (1, 'The Night Land', 'William Hope Hodgson');
INSERT INTO Books VALUES (2, 'Vathek', 'William Beckford');
INSERT INTO Books VALUES (3, 'The King in Yellow', 'Robert W. Chambers');
INSERT INTO Books VALUES (4, 'Flatland', 'Edwin A. Abbott');
INSERT INTO Books VALUES (5, 'Rasselas', 'Samuel Johnson');

-- c. Create an index on the Title column
CREATE INDEX idx_title ON Books (Title);

-- Query to select books by a specific author
SELECT * FROM Books WHERE Author = 'George Orwell';
```


# Q2

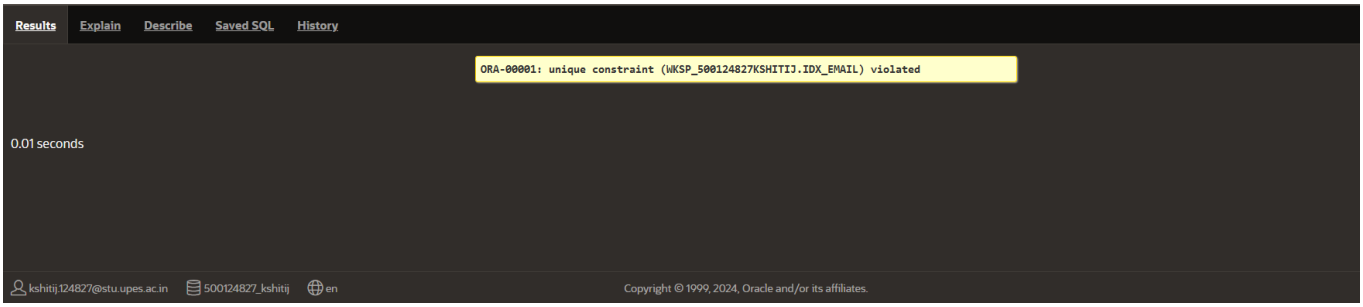
```sql
CREATE TABLE Employees ( EmployeeID NUMBER PRIMARY KEY, Email VARCHAR2(100),
Department VARCHAR2(50) );

INSERT ALL
INTO Employees (EmployeeID, Email, Department) VALUES (1,
'elias.hart@Google.com', 'HR')
INTO Employees (EmployeeID, Email, Department) VALUES (2,
'matilda.rose@Google.com', 'IT')
INTO Employees (EmployeeID, Email, Department) VALUES (3,
'quincy.lane@Google.com', 'Finance')
```

```
INTO Employees (EmployeeID, Email, Department) VALUES (4,
'opal.moon@Google.com', 'Marketing')
INTO Employees (EmployeeID, Email, Department) VALUES (5,
'finn.bishop@Google.com', 'Sales')
SELECT * FROM dual;

CREATE UNIQUE INDEX idx_email ON Employees (Email);

INSERT INTO Employees (EmployeeID, Email, Department) VALUES (6,
'opal.moon@Google.com', 'Legal');
```




# Q3

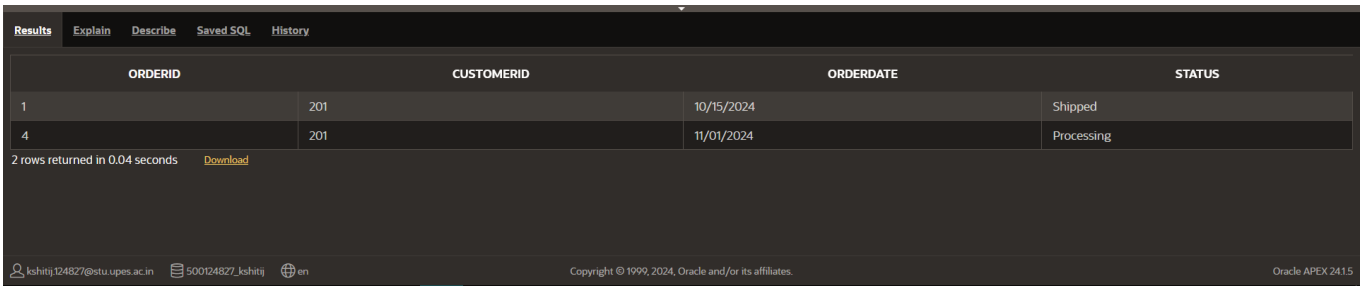
```
-- a. Create the Orders table
CREATE TABLE Orders (
    OrderID NUMBER PRIMARY KEY,
    CustomerID NUMBER,
    OrderDate DATE,
    Status VARCHAR2(20)
);

-- b. Insert 5 records into the Orders table
INSERT INTO Orders (OrderID, CustomerID, OrderDate, Status) VALUES (1, 201,
TO_DATE('2024-10-15', 'YYYY-MM-DD'), 'Shipped');
INSERT INTO Orders (OrderID, CustomerID, OrderDate, Status) VALUES (2, 202,
TO_DATE('2024-10-18', 'YYYY-MM-DD'), 'Pending');
INSERT INTO Orders (OrderID, CustomerID, OrderDate, Status) VALUES (3, 203,
TO_DATE('2024-10-20', 'YYYY-MM-DD'), 'Delivered');
INSERT INTO Orders (OrderID, CustomerID, OrderDate, Status) VALUES (4, 201,
TO_DATE('2024-11-01', 'YYYY-MM-DD'), 'Processing');
INSERT INTO Orders (OrderID, CustomerID, OrderDate, Status) VALUES (5, 204,
TO_DATE('2024-11-05', 'YYYY-MM-DD'), 'Cancelled');

-- c. Create a composite index on CustomerID and OrderDate
CREATE INDEX idx_customer_orderdate ON Orders (CustomerID, OrderDate);

-- Query to select orders by a specific customer and date range
```

```sql
SELECT *
FROM Orders
WHERE CustomerID = 201
AND OrderDate BETWEEN TO_DATE('2024-10-01', 'YYYY-MM-DD') AND TO_DATE('2024-11-15', 'YYYY-MM-DD');
```



Results | Explain | Describe | Saved SQL | History

| ORDERID | CUSTOMERID | ORDERDATE | STATUS |
|---|---|---|---|
| 1 | 201 | 10/15/2024 | Shipped |
| 4 | 201 | 11/01/2024 | Processing |

2 rows returned in 0.04 seconds Download

kshitij.124827@stu.upes.ac.in 500124827_kshitij en Copyright © 1999, 2024, Oracle and/or its affiliates. Oracle APEX 24.1.5

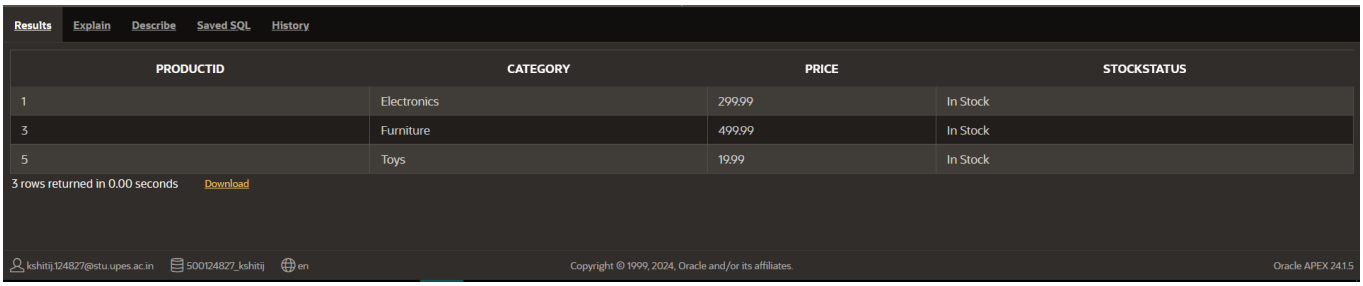
# Q4

```sql
-- a. Create the Products table
CREATE TABLE Products (
    ProductID NUMBER PRIMARY KEY,
    Category VARCHAR2(50),
    Price NUMBER,
    StockStatus VARCHAR2(20)
);

-- b. Insert 5 records into the Products table
INSERT INTO Products (ProductID, Category, Price, StockStatus) VALUES (1, 'Electronics', 299.99, 'In Stock');
INSERT INTO Products (ProductID, Category, Price, StockStatus) VALUES (2, 'Home Appliances', 149.99, 'Out of Stock');
INSERT INTO Products (ProductID, Category, Price, StockStatus) VALUES (3, 'Furniture', 499.99, 'In Stock');
INSERT INTO Products (ProductID, Category, Price, StockStatus) VALUES (4, 'Clothing', 39.99, 'Low Stock');
INSERT INTO Products (ProductID, Category, Price, StockStatus) VALUES (5, 'Toys', 19.99, 'In Stock');

-- c. Create a bitmap index on the StockStatus column
CREATE BITMAP INDEX idx_stockstatus ON Products (StockStatus);

-- d. Write a query to select products by stock status
SELECT *
FROM Products
WHERE StockStatus = 'In Stock';
```



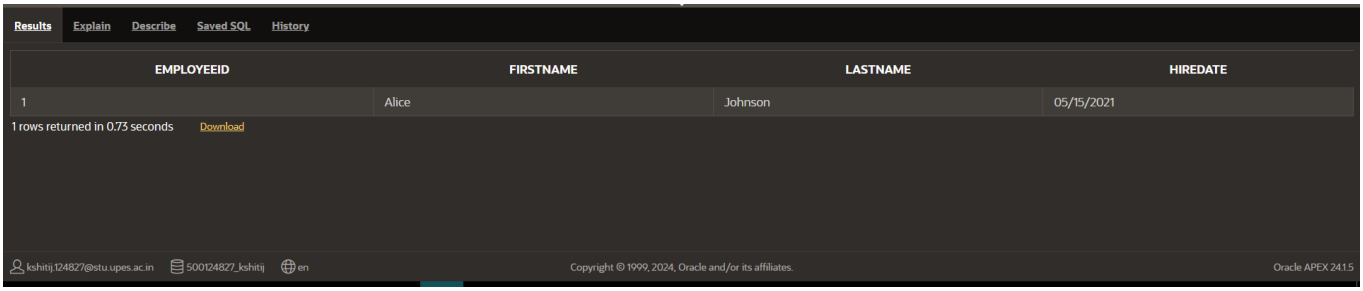

# Q5

```sql
-- a. Create the Employees table
CREATE TABLE Employees (
    EmployeeID NUMBER PRIMARY KEY,
    FirstName VARCHAR2(50),
    LastName VARCHAR2(50),
    HireDate DATE
);

-- b. Insert 5 records into the Employees table
INSERT INTO Employees (EmployeeID, FirstName, LastName, HireDate) VALUES (1,
'Alice', 'Johnson', TO_DATE('2021-05-15', 'YYYY-MM-DD'));
INSERT INTO Employees (EmployeeID, FirstName, LastName, HireDate) VALUES (2,
'Bob', 'Smith', TO_DATE('2019-03-10', 'YYYY-MM-DD'));
INSERT INTO Employees (EmployeeID, FirstName, LastName, HireDate) VALUES (3,
'Charlie', 'Brown', TO_DATE('2020-07-22', 'YYYY-MM-DD'));
INSERT INTO Employees (EmployeeID, FirstName, LastName, HireDate) VALUES (4,
'Diana', 'Clark', TO_DATE('2022-01-05', 'YYYY-MM-DD'));
INSERT INTO Employees (EmployeeID, FirstName, LastName, HireDate) VALUES (5,
'Ethan', 'White', TO_DATE('2018-11-30', 'YYYY-MM-DD'));

-- c. Create a function-based index on the concatenation of FirstName and
LastName
CREATE INDEX idx_fullname ON Employees (UPPER(FirstName || ' ' || LastName));

-- d. Write a query to select employees by full name
SELECT *
FROM Employees
WHERE UPPER(FirstName || ' ' || LastName) = UPPER('Alice Johnson');
```




# Q6

```sql
-- a. Create the Sales table with a virtual column TaxAmount
CREATE TABLE Sales (
    SaleID NUMBER PRIMARY KEY,
    SaleAmount NUMBER,
    TaxAmount AS (SaleAmount * 0.1)  -- Virtual column for 10% of SaleAmount
);

-- b. Insert 5 records into the Sales table
INSERT INTO Sales (SaleID, SaleAmount) VALUES (1, 500);
INSERT INTO Sales (SaleID, SaleAmount) VALUES (2, 1200);
INSERT INTO Sales (SaleID, SaleAmount) VALUES (3, 750);
INSERT INTO Sales (SaleID, SaleAmount) VALUES (4, 2000);
INSERT INTO Sales (SaleID, SaleAmount) VALUES (5, 300);

-- c. Create an index on the virtual column TaxAmount
CREATE INDEX idx_taxamount ON Sales (TaxAmount);

-- d. Write a query to select sales where TaxAmount is greater than a certain value
SELECT *
FROM Sales
WHERE TaxAmount > 100;
```