PRODUCT SUPPLY MANAGEMENT SYSTEM

Submitted as a part of course curriculum for

UE19CS301 DATABASE MANAGEMENT SYSTEM

Under the guidance of,

Prof. Nivedita Kasturi
Assistant Professor

Name	SRN
Anagha Suresh	PES2UG19CS037

Simple queries

1. A query to retrieve product id and product description of type = 'Food items'

```
productsupply=# select product.pro_id,pro_desc
productsupply-# from Product,Producttype
productsupply-# where pro_type='Food items' AND producttype.pro_id=product.pro_id;
 pro_id |    pro_desc
--------+----------------
     22 | Ferrero Rocher
(1 row)
```

2. Retrieve user email as mail_id, user's address as address and mobile number as ph_no of users_id = 789

```
productsupply=# select user_email as mail_id, user_address as address, user_mobile as ph_no
productsupply-# from users
productsupply-# where users_id=789;
     mail_id        | address |   ph_no
--------------------+---------+------------
 kiarar01@gmail.com | ABC     | 9513236789
(1 row)
```

3. Retrieve role name of an user named 'Charlotte'

```
productsupply=# SELECT Role_name
productsupply-# FROM Roles,Users
productsupply-# WHERE User_fname='Charlotte' AND Usernum=Users_id;
 role_name
-----------
 Supplier2
(1 row)
```

4. Display product description as Product and product number as availability from product table ordered by product number

```
productsupply=# SELECT Pro_desc as Product,Pro_num as Availability
productsupply-# FROM Product
productsupply-# ORDER BY Pro_num;
        product         | availability
------------------------+--------------
 Mahavir Oil            |            2
 Lenovo Laptop          |            7
 Harry Potter           |            8
 HP Desktop             |           11
 Ferrero Rocher         |           12
 Eureka Forbes  cleaner |           13
 cotton raw material    |           23
 Aidea Microfibre cloth |           54
 Wings of Fire          |           62
 Versace Crystal        |           90
 Lakhme Eyeliner        |          111
(11 rows)
```

5. Retrieve delivery description as product_to_deliver in between the delivery date 2-10-2021 and 12-10-2021

```
productsupply=# SELECT Del_desc as Product_to_deliver
productsupply-# FROM Delivery
productsupply-# WHERE Del_date BETWEEN '2021-10-2' AND '2021-10-12';
 product_to_deliver
--------------------
 electronic devices
 cosmetics
(2 rows)
```

Complex Queries

1. Display distinct pro_type of those products that are less than 20

```
productsupply=# SELECT DISTINCT Pro_type as products_less_than_20
productsupply-# FROM Producttype
productsupply-# WHERE EXISTS (SELECT Pro_desc
productsupply(# FROM Product
productsupply(# WHERE Producttype.Pro_id=Product.Pro_id AND Pro_num < 20);
 products_less_than_20
-----------------------
 electronic device
 Cleaning supply
 Food items
 Book
 Supplies
(5 rows)
```

2. Display the permission names given to the user with per_id 10,14, and 15

```
productsupply=# SELECT Per_name
productsupply-# FROM Permission
productsupply-# WHERE Per_id IN (10,14,15);
 per_name
-----------
 Supplier1
 Supplier2
 Supplier3
(3 rows)
```

3. Display user_id, user_fname, user_lname and role_name from table users joined with roles

```
productsupply=# SELECT usr.users_id,usr.user_fname,usr.user_lname,role.role_name
productsupply-# FROM Users as usr
productsupply-# NATURAL JOIN Roles as role
productsupply-# WHERE Usernum=Users_id;
 users_id | user_fname | user_lname | role_name
----------+------------+------------+-----------
      113 | Ashish     | Patel      | Admin
      456 | Manas      | Bhat       | Supplier1
      789 | Kiara      | Ram        | Manager
      191 | Charlotte  | Steven     | Supplier2
      112 | Prateek    | Meher      | Supplier3
(5 rows)
```

4. Retrieve the count of stk_type grouped by stk_type

```
productsupply=# SELECT COUNT(Stk_type), Stk_type
productsupply-# FROM Stocktype
productsupply-# GROUP BY Stk_type;
 count |      stk_type
-------+--------------------
     2 | cleaning supplies
     2 | electronic device
     1 | Raw materials
     1 | oil-supplies
     1 | cosmetics
     2 | Books
(6 rows)
```

5. Create a column named quantityconclusion which displays 'The quantity is less than 50' if pro_num less than 50 and 'The quantity is more than 50' if pro_num is greater than and 'The quantity is 50' otherwise

```
productsupply=# SELECT Pro_id,Pro_num,
productsupply-# CASE
productsupply-# WHEN Pro_num > 50 THEN 'The quantity is greater than 50'
productsupply-# WHEN Pro_num < 50 THEN 'The quantity is less than 50'
productsupply-# ELSE 'The quantity is 50'
productsupply-# END AS QuantityConclusion
productsupply-# FROM Product;
 pro_id | pro_num |         quantityconclusion
--------+---------+------------------------------------
      1 |      11 | The quantity is less than 50
      3 |       7 | The quantity is less than 50
      5 |       8 | The quantity is less than 50
      9 |      62 | The quantity is greater than 50
     10 |      23 | The quantity is less than 50
     22 |      12 | The quantity is less than 50
     24 |      13 | The quantity is less than 50
     29 |      54 | The quantity is greater than 50
     25 |     111 | The quantity is greater than 50
     27 |      90 | The quantity is greater than 50
     88 |       2 | The quantity is less than 50
(11 rows)
```

Access privileges

Creating users: Admin, manager and suppliers

```
productsupply=# CREATE USER Admin;
CREATE ROLE
productsupply=# CREATE USER Manager;
CREATE ROLE
productsupply=# CREATE USER Supplier1;
CREATE ROLE
productsupply=# CREATE USER Supplier2;
CREATE ROLE
productsupply=# CREATE USER Supplier3;
CREATE ROLE
productsupply=#
```

List of users and granting all access privileges to Admin

```
productsupply=# SELECT usename from pg_user;
  usename
-----------
 postgres
 user20
 admin
 manager
 supplier1
 supplier2
 supplier3
(7 rows)


productsupply=# \du
                                  List of roles
 Role name |                        Attributes                        | Member of
-----------+----------------------------------------------------------+-----------
 admin     |                                                          | {}
 manager   |                                                          | {}
 postgres  | Superuser, Create role, Create DB, Replication, Bypass RLS | {}
 supplier1 |                                                          | {}
 supplier2 |                                                          | {}
 supplier3 |                                                          | {}
 user20    |                                                          | {}


productsupply=# GRANT ALL ON DATABASE productsupply TO admin;
GRANT
```

Granting access to suppliers and manager

```
productsupply=# GRANT INSERT,DELETE,UPDATE,SELECT ON Inventory,Invitems,Invtype,Stock,Stocktype TO manager;
GRANT
productsupply=# GRANT INSERT,DELETE ON Product,Producttype TO supplier1;
GRANT
```

```
productsupply=# GRANT INSERT,DELETE,SELECT ON Product,Producttype TO supplier2;
GRANT
productsupply=# GRANT INSERT,DELETE ON Product,Producttype TO supplier3;
GRANT
```

Revoking a permission from supplier2 and list of permissions to different tables

```
productsupply=# REVOKE SELECT ON Product,Producttype FROM supplier2;
REVOKE
productsupply=# \z
                                  Access privileges
 Schema |      Name       | Type  |      Access privileges      | Column privileges | Policies
--------+-----------------+-------+-----------------------------+-------------------+----------
 public | delivery        | table |                             |                   |
 public | delivery_type   | table |                             |                   |
 public | expressdelivery | table |                             |                   |
 public | inventory       | table | postgres=arwdDxt/postgres+  |                   |
        |                 |       | manager=arwd/postgres       |                   |
 public | invitems        | table | postgres=arwdDxt/postgres+  |                   |
        |                 |       | manager=arwd/postgres       |                   |
 public | invtype         | table | postgres=arwdDxt/postgres+  |                   |
        |                 |       | manager=arwd/postgres       |                   |
 public | login           | table |                             |                   |
 public | membership      | table |                             |                   |
 public | permission      | table |                             |                   |
 public | product         | table | postgres=arwdDxt/postgres+  |                   |
        |                 |       | supplier1=ad/postgres    +  |                   |
        |                 |       | supplier2=ad/postgres    +  |                   |
        |                 |       | supplier3=ad/postgres       |                   |
 public | producttype     | table | postgres=arwdDxt/postgres+  |                   |
        |                 |       | supplier1=ad/postgres    +  |                   |
        |                 |       | supplier2=ad/postgres    +  |                   |
        |                 |       | supplier3=ad/postgres       |                   |
 public | roles           | table |                             |                   |
 public | stock           | table | postgres=arwdDxt/postgres+  |                   |
        |                 |       | manager=arwd/postgres       |                   |
 public | stocktype       | table | postgres=arwdDxt/postgres+  |                   |
        |                 |       | manager=arwd/postgres       |                   |
 public | users           | table |                             |                   |
(15 rows)
```

Testing the granted permissions by entering postgreSQL, as supplier1

```
C:\Program Files\PostgreSQL\13\bin>psql -d productsupply -U supplier1
Password for user supplier1:
psql (13.4)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

productsupply=> SELECT * FROM Product;
ERROR:  permission denied for table product
productsupply=>
```

As manager: can insert into inventory but cannot select from product table

```
C:\Program Files\PostgreSQL\13\bin>psql -d productsupply -U manager
Password for user manager:
psql (13.4)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

productsupply=> SELECT * FROM Stock;
 stk_id |        stk_desc        | iid
--------+------------------------+-----
   1230 | Lenovo Laptop          | 200
   1234 | Hp desktop             | 200
   1236 | Harry Potter           | 202
   1237 | Wings of Fire          | 202
   1232 | Cotton Raw material    | 204
   1238 | Lakhme Eyeliner        | 207
   1239 | Eureka Forbes cleaner  | 205
   1244 | Aidea Microfibre Cloth | 205
   1233 | Mahavir oil            | 210
(9 rows)


productsupply=> SELECT * FROM Product;
ERROR:  permission denied for table product
```

```
productsupply=> INSERT INTO Inventory VALUES(208,18,'home supplies');
INSERT 0 1
```