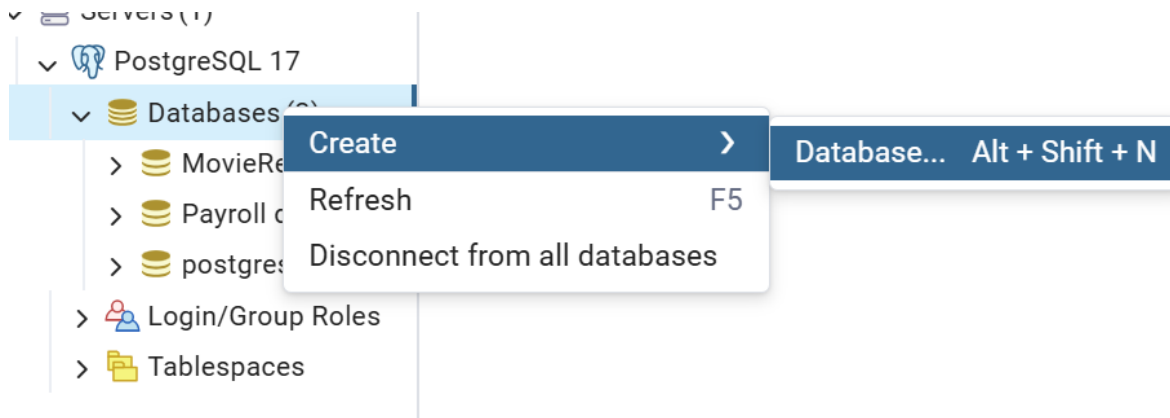
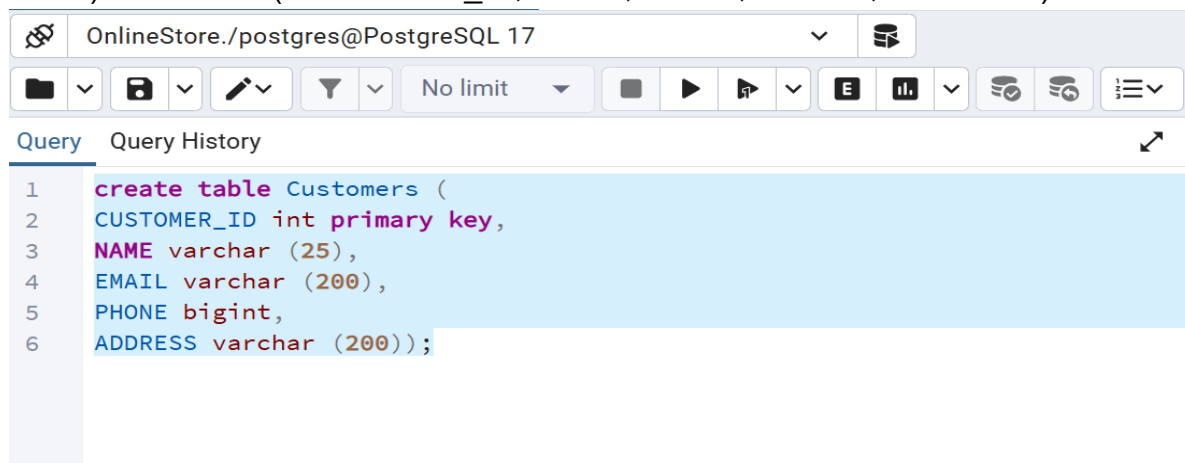


3.1 Database Creation: Create a database named OnlineStore.



Create tables:

3.1 a) Customers (CUSTOMER_ID, NAME, EMAIL, PHONE, ADDRESS)

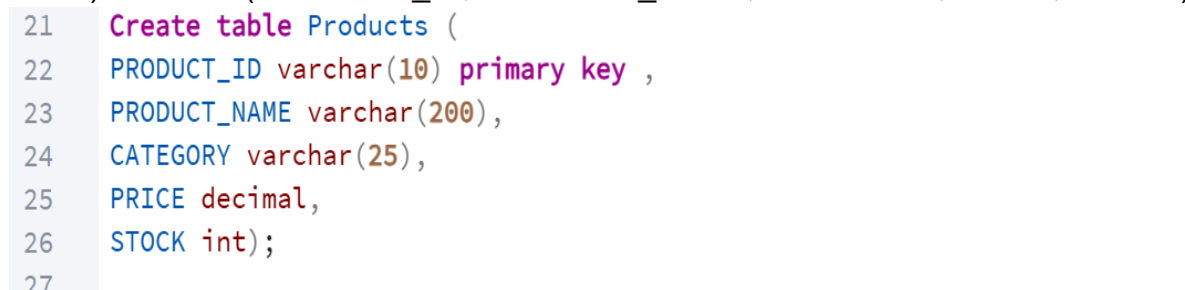


Data Output Messages Notifications

CREATE TABLE

Query returned successfully in 85 msec.

3.1 b) Products (PRODUCT_ID, PRODUCT_NAME, CATEGORY, PRICE, STOCK)



Data Output Messages Notifications

CREATE TABLE

Query returned successfully in 59 msec.

3.1 c) Orders (ORDER_ID, CUSTOMER_ID, PRODUCT_ID, QUANTITY, ORDER_DATE)

```
28 Create table Orders (  
29 ORDER_ID int,  
30 CUSTOMER_ID int,  
31 PRODUCT_ID varchar(10),  
32 QUANTITY int,  
33 ORDER_DATE DATE,  
34 foreign key (CUSTOMER_ID) references Customers(CUSTOMER_ID),  
35 foreign key (PRODUCT_ID) references Products(PRODUCT_ID));
```

Data Output Messages Notifications

CREATE TABLE

Query returned successfully in 86 msec.

3.2 Data Creation:

Insert sample data for customers.

```
8 Insert into Customers values  
9 (211, 'Shankar', 'Shankar@gmail.com', 9176221122, 'Madavaram, Chennai'),  
10 (222, 'Mukesh', 'Mukesh@gmail.com', 9176220413, 'Kodambakkam, Chennai'),  
11 (233, 'Praveen', 'Praveen@gmail.com', 9176220474, 'Anna Nagar, Chennai'),  
12 (244, 'Shrinath', 'Shrinath@gmail.com', 9176110415, 'Saidapet, Chennai'),  
13 (255, 'Vignesh', 'Vignesh@gmail.com', 9176239116, 'velachery, Chennai'),  
14 (266, 'Vimalrka', 'Vimalrka@gmail.com', 9135674171, 'Tambaram, Chennai'),  
15 (277, 'Kamaraj', 'Kamaraj@gmail.com', 9176220418, 'Medavakkam, Chennai'),  
16 (288, 'Sekar', 'Sekar@gmail.com', 9176231418, 'Anna Nagar, Chennai'),  
17 (299, 'Lokesh', 'Lokesh@gmail.com', 9176276751, 'Anna Nagar, Chennai'),  
18 (311, 'Joshua', 'Joshua@gmail.com', 9176220498, 'velachery, Chennai');
```

Data Output Messages Notifications

INSERT 0 10

Query returned successfully in 151 msec.

Insert sample data for products

```
36  
37 Insert into Products values  
38 ('B0CK2QHML3', 'SKY-TOUCH Dog Treats: 400g Chicken Wrapped', 'PET_FOOD', 21  
39 ('B0C3RJVT3M', 'SKY-TOUCH Auto Close Safety Baby Gate, Extra Wide Child G  
40 ('B0B9J4577K', 'SKY-TOUCH Baby Proofing Plug Covers, Plug Socket Covers, Ou  
41 ('B0DHLJDDJ3', 'SKY-TOUCH 80 Pieces Baby Disposable Changing Mats, Waterpr  
42 ('B0CJM5X73Z', 'SKY-TOUCH Auto Close Safety Baby Gate', 'TEMPORARY_GATE', 9  
43 ('B0CSK48PWX', 'SKY-TOUCH Auto Close Safety Baby Gate, 80.00 x 6.00 x 75.  
44 ('B09QQQVGT3', 'SKY TOUCH 5pcs Muslin Baby Washcloths and Towels, Natural  
45 ('B0DJ6TT47R', 'SKY-TOUCH Handprint and Footprint Frame Makers Kit, Newbo  
46 ('B0CKXCDW97', 'SKY-TOUCH Auto Close Safety Baby Gate, Extra Wide Child G
```

Data Output Messages Notifications

INSERT 0 9

Query returned successfully in 81 msec.

Insert sample data for orders.

```
48 Insert into Orders values
49 (101,211,'B09QQQVGT3',1,'2025-04-25'),
50 (102,222,'B0DJ6TT47R',2,'2025-05-26'),
51 (103,233,'B0C3RJVT3M',4,'2025-07-27'),
52 (104,244,'B0B9J4577K',5,'2025-04-21'),
53 (105,255,'B0DHLJDDJ3',3,'2025-04-22'),
54 (106,266,'B0C3RJVT3M',2,'2025-07-28'),
55 (107,277,'B0CSK48PWX',1,'2025-05-31'),
56 (108,288,'B0C3RJVT3M',2,'2025-03-18'),
57 (109,299,'B0B9J4577K',3,'2025-02-08'),
58 (110,311,'B0C3RJVT3M',4,'2025-05-30');
```

Data Output Messages Notifications

INSERT 0 10

Query returned successfully in 90 msec.

3.3 Order Management:

3.3 a) Retrieve all orders placed by a specific customer.

```
60 --Order Management:
61 --3.3 a) Retrieve all orders placed by a specific customer.*/
62
63 select *, (select c.NAME
64 from Customers c
65 where c.CUSTOMER_ID = o.CUSTOMER_ID) as Customer_Name
66 from orders o;
```

Data Output Messages Notifications

	order_id integer	customer_id integer	product_id character varying	quantity integer	order_date date	customer_name character varying (2
1	101	211	B09QQQVGT3	1	2025-04-25	Shankar
2	102	222	B0DJ6TT47R	2	2025-05-26	Mukesh
3	103	233	B0C3RJVT3M	4	2025-07-27	Praveen
4	104	244	B0B9J4577K	5	2025-04-21	Shrinath
5	105	255	B0DHLJDDJ3	3	2025-04-22	Vignesh
6	106	266	B0C3RJVT3M	2	2025-07-28	Vimalrka

3.3 b) Find products that are out of stock.

```
68 --Find products that are out of stock.
69
70 select stock,product_id
71 from products
72 where stock <=1; --consider stock less than or equal to 1 is out of stock.
```

Data Output Messages Notifications

Showing rows: 1 to 1 Page No: 1

	stock integer	product_id [PK] character varying (10)
1	1	B0DJ6TT47R

3.3 c) Calculate the total revenue generated per product.

```
74 --3.3 c) Calculate the total revenue generated per product.
75
76 select P.product_id,P.PRICE,SUM(P.PRICE * o.QUANTITY) as Total_revenue
77 from products p
78 join orders o on p.PRODUCT_ID = o.PRODUCT_ID
79 GROUP BY p.product_id, p.PRODUCT_NAME;
80
```

Data Output Messages Notifications

Showing rows: 1 to 6 Page No: 1

	product_id [PK] character varying (10)	price numeric	total_revenue numeric
1	B0DJ6TT47R	27.9	55.8
2	B0CSK48PWX	96.2	96.2
3	B0B9J4577K	14.5	116.0
4	B0DHLJDDJ3	56.3	168.9
5	B0C3RJVT3M	127.6	1531.2
6	B09QQQVGT3	18.1	18.1

Total rows: 6 Query complete 00:00:00.243

3.3 d) Retrieve the top 5 customers by total purchase amount.

```

81  --3.3 d)Retrieve the top 5 customers by total purchase amount.
82  select c.NAME,P.product_id,P.PRICE,sum(P.PRICE * o.QUANTITY) as Total_revenue
83  from products p
84  join orders o on p.PRODUCT_ID = o.PRODUCT_ID
85  join customers c on c.CUSTOMER_ID = o.CUSTOMER_ID
86  GROUP BY p.product_id, p.PRODUCT_NAME,c.NAME
87  order by Total_revenue desc
88  limit 5;

```

Data Output Messages Notifications

Showing rows: 1 to 5 Page No: 1

	name character varying (25)	product_id character varying (10)	price numeric	total_revenue numeric
1	Joshua	B0C3RJVT3M	127.6	510.4
2	Praveen	B0C3RJVT3M	127.6	510.4
3	Sekar	B0C3RJVT3M	127.6	255.2
4	Vimalrka	B0C3RJVT3M	127.6	255.2
5	Vignesh	B0DHLJDDJ3	56.3	168.9

Total rows: 5 Query complete 00:00:00.152

3.3 e) Find customers who placed orders in at least two different product categories.

```

91  --3.3 e) Find customers who placed orders in at least two different product categories.
92
93  select c.NAME,P.product_id
94  from products p
95  join orders o on p.PRODUCT_ID = o.PRODUCT_ID
96  join customers c on c.CUSTOMER_ID = o.CUSTOMER_ID
97  GROUP BY c.NAME,P.product_id
98  having count(DISTINCT P.product_id) >= 2;

```

Data Output Messages Notifications

name character varying (25)	product_id character varying (10)
--------------------------------	--------------------------------------

3.4 Analytics:

3.4 a) Find the month with the highest total sales.

```

100  --3.4 Analytics:
101  --3.4 a) Find the month with the highest total sales
102
103  select TO_CHAR(ORDER_DATE, 'Month') as Months,
104         sum(p.price * o.QUANTITY) as total_sales
105  from orders o
106  join products p on p.PRODUCT_ID = o.PRODUCT_ID
107  group by Months
108  order by total_sales desc
109  LIMIT 1;

```

Data Output Messages Notifications

	months text	total_sales numeric
1	July	765.6

3.4 b) Identify products with no orders in the last 6 months.

```

111  --3.4 b) Identify products with no orders in the last 6 months.
112
113  select p.product_id, p.product_name, o.ORDER_ID
114  from products p
115  left join orders o on p.product_id = o.product_id
116     and o.order_date >= CURRENT_DATE - INTERVAL '6 months'
117  where o.order_id IS NULL;

```

Data Output Messages Notifications

	product_id character varying (10)	product_name character varying (200)	order_id integer
1	B0CJM5X73Z	SKY-TOUCH Auto Close Safety Baby Gate	[null]
2	B0CKXCDW97	SKY-TOUCH Auto Close Safety Baby Gate, Extra Wide Child Gate 75-84cm + 10cm	[null]
3	B0CK2QHML3	SKY-TOUCH Dog Treats: 400g Chicken Wrapped	[null]

3.4 c) Retrieve customers who have never placed an order.

```

119  --3.4 c) Retrieve customers who have never placed an order.
120
121  select c.name, o.ORDER_ID
122  from customers c
123  left join orders o on c.CUSTOMER_ID = o.CUSTOMER_ID
124  where o.order_id is null;

```

Data Output Messages Notifications

	name character varying (25)	order_id integer
--	--------------------------------	---------------------

3.4 d) Calculate the average order value across all orders.

```
126 --3.4 d) Calculate the average order value across all orders.
127 SELECT AVG(OrderTotal.TotalOrderValue) AS AverageOrderValue
128 FROM
129     (SELECT o.order_id, SUM(p.price * o.quantity) AS TotalOrderValue
130     FROM orders o
131     JOIN products p ON o.product_id = p.product_id
132     GROUP BY o.order_id) AS OrderTotal;
```

Data Output Messages Notifications

Showing rows: 1 to 1

Page No: 1

averageordervalue
numeric

1 198.6200000000000000