












# SQL Queries with Images and Explanations

12  select \* from sales\_data

13 where orderid >1005;

Data Output Messages Notifications




	orderid integer	customername text	product text	category text	quantity integer	price numeric (10,2)	orderdate date	region text
1	1006	Linda Martinez	Printer	Electronics	1	150.00	2024-01-12	South
2	1007	Michael Garcia	Bookshelf	Furniture	2	90.00	2024-01-13	East
3	1008	Barbara Anderson	Tablet	Electronics	1	300.00	2024-01-14	West
4	1009	William Thomas	Monitor	Electronics	2	180.00	2024-01-15	North
5	1010	Elizabeth Taylor	Mouse	Electronics	5	20.00	2024-01-16	South

This query orders all rows from the 'sales\_data' table in ascending order of 'orderid'. It ensures the results are displayed starting from the lowest orderid to the highest.

14











15

16  SELECT region, SUM(price) AS total\_sales

17 FROM sales\_data

18 GROUP BY region;

Data Output Messages Notifications



	region text	total_sales numeric
1	East	210.00
2	South	670.00
3	West	500.00
4	North	1010.00

An INNER JOIN between 'sales\_data' and 'customer' retrieves only records with matching 'customername' values in both tables.

```

72
73 select * from sales_data
74 order by orderid asc;
75
76

```

	orderid integer	customername text	product text	category text	quantity integer	price numeric (10,2)	orderdate date	region text
1	1001	Zeeshan Ansari	Laptop	Electronics	1	500.00	2024-01-05	North
2	1002	Mary Johnson	Smartphone	Electronics	2	600.00	2024-01-06	South
3	1003	James Brown	Office Chair	Furniture	1	150.00	2024-01-07	East
4	1004	Patricia Davis	Desk	Furniture	1	300.00	2024-01-08	West
5	1005	Robert Wilson	Headphones	Electronics	3	120.00	2024-01-10	North
6	1006	Linda Martinez	Printer	Electronics	1	200.00	2024-01-12	South
7	1007	Michael Garcia	Bookshelf	Furniture	2	100.00	2024-01-13	East
8	1008	Barbara Anderson	Tablet	Electronics	1	240.00	2024-01-14	West

A LEFT JOIN is used here, retrieving all records from 'sales\_data' and matching data from 'customer'. Rows with no matching customer have NULL values in 'customer' table columns.

```

81
82 select * from sales_data as s
83 inner join customer as c
84 on s.customername= c.customername;

```

	orderid integer	customername text	product text	category text	quantity integer	price numeric (10,2)	orderdate date	region text	customername character varying (250)
1	1005	Robert Wilson	Headphones	Electronics	3	120.00	2024-01-10	North	Robert Wilson
2	1007	Michael Garcia	Bookshelf	Furniture	2	100.00	2024-01-13	East	Michael Garcia
3	1001	Zeeshan Ansari	Laptop	Electronics	1	500.00	2024-01-05	North	Zeeshan Ansari

A RIGHT JOIN retrieves all records from 'customer' and matching rows from 'sales\_data'. If no match exists, NULL values appear in 'sales\_data' columns.

```

85
86 v select * from sales_data as s
87 left join customer as c
88 on s.customername=c.customername;

```

Data Output Messages Notifications

	orderid integer	customername text	product text	category text	quantity integer	price numeric (10,2)	orderdate date	region text	customername character varying (250)
4	1005	Robert Wilson	Headphones	Electronics	3	120.00	2024-01-10	North	Robert Wilson
5	1006	Linda Martinez	Printer	Electronics	1	200.00	2024-01-12	South	[null]
6	1007	Michael Garcia	Bookshelf	Furniture	2	100.00	2024-01-13	East	Michael Garcia
7	1008	Barbara Anderson	Tablet	Electronics	1	340.00	2024-01-14	West	[null]
8	1009	William Thomas	Monitor	Electronics	2	200.00	2024-01-15	North	[null]
9	1010	Elizabeth Taylor	Mouse	Electronics	5	50.00	2024-01-16	South	[null]
10	1001	Zeeshan Ansari	Laptop	Electronics	1	500.00	2024-01-05	North	Zeeshan Ansari

A subquery filters rows where the 'price' is greater than the average price in the 'sales\_data' table.

```

85
86 v select * from sales_data as s
87 right join customer as c
88 on s.customername=c.customername;

```

Data Output Messages Notifications

	orderid integer	customername text	product text	category text	quantity integer	price numeric (10,2)	orderdate date	region text	customername character varying (250)
1	1005	Robert Wilson	Headphones	Electronics	3	120.00	2024-01-10	North	Robert Wilson
2	1007	Michael Garcia	Bookshelf	Furniture	2	100.00	2024-01-13	East	Michael Garcia
3	1001	Zeeshan Ansari	Laptop	Electronics	1	500.00	2024-01-05	North	Zeeshan Ansari
4	[null]	[null]	[null]	[null]	[null]	[null]	[null]	[null]	John Smith

Calculates the average price using AVG() and rounds the result to 2 decimal places.

```

93
94  SELECT customername, price
95  FROM sales_data
96  WHERE price > (
97      SELECT AVG(price) FROM sales_data
98  );
99

```

Data Output Messages Notifications

	customername text	price numeric (10,2)
1	Mary Johnson	600.00
2	Patricia Davis	300.00
3	Barbara Anderson	340.00
4	Zeeshan Ansari	500.00

Calculates the total sum of the 'price' column from the 'sales\_data' table using SUM().

```

100
101  select round(avg(price),2) from sales_data;

```

Data Output Messages Notifications

	round numeric
1	256.00

Creates a view 'region\_sales' that groups 'sales\_data' by region and calculates total sales for each.

VS

```

100
101 select sum(price) from sales_data;

```

Data Output Messages Notifications

	sum numeric
1	2560.00

S

Groups 'sales\_data' by 'region' and calculates total sales using SUM(price). Displays the sales per region.

```

102
103 CREATE VIEW region_sales AS
104 SELECT region, SUM(price) AS total_sales
105 FROM sales_data
106 GROUP BY region;
107
108 SELECT * FROM region_sales;

```

Data Output Messages Notifications

	region text	total_sales numeric
1	East	250.00
2	South	850.00
3	West	640.00
4	North	820.00

Filters and retrieves records where 'orderid' is greater than 1005 from 'sales\_data'.