

1.

```
1 SELECT customer_id, customer_city, customer_state  
2 FROM olist_customers_dataset;
```

	customer_id	customer_city	customer_state
1	06b8999e2f...	franca	SP
2	18955e83d3...	sao ...	SP
3	4e7b3e0028...	sao paulo	SP
4	b2b6027bc5...	mogi das ...	SP
5	4f2d8ab171...	campinas	SP
6	879864dab9...	jaragua do...	SC
7	fd826e7cf6...	sao paulo	SP
8	5e274e7a0c...	timoteo	MG
9	5adf08e34b...	curitiba	PR
10	4b7139f345...	belo ...	MG
11	9fb35e4ed6...	montes ...	MG
12	5aa9e4fdd4...	rio de ...	RJ

Execution finished without errors.

Result: 99441 rows returned in 164ms

At line 1:

```
SELECT customer_id, customer_city, customer_state  
FROM olist_customers_dataset;
```

2.

```
1 SELECT customer_id, customer_city, customer_state  
2 FROM olist_customers_dataset  
3 WHERE customer_state = 'SP';
```

	customer_id	customer_city	customer_state
1	06b8999e2fbalalfbc88172c00ba8bc7	franca	SP
2	18955e83d337fd6b2def6b18a428ac77	sao bernardo do campo	SP
3	4e7b3e00288586ebd08712fdd0374a03	sao paulo	SP
4	b2b6027bc5c5109e529d4dc6358b12c3	mogi das cruzes	SP
5	4f2d8ab171c80ec8364f7c12e35b23ad	campinas	SP
6	fd826e7cf63160e536e0908c76c3f41	sao paulo	SP
7	b2d1536598b73a9abd18e0d75d92f0a3	lencois paulista	SP
8	eabebad39a88bb6f5b52376faec28612	sao paulo	SP
9	206f3129c0e4d7d0b9550426023f0a08	piracicaba	SP
10	c5c61596a3b6bd0cee5766992c48a9a1	guarulhos	SP
11	9b8ce803689b3562defaad4613ef426f	sao paulo	SP
12	2938121a40a20953c43caa8c98787fcb	sao paulo	SP

Execution finished without errors.

Result: 41746 rows returned in 268ms

At line 1:

```
SELECT customer_id, customer_city, customer_state  
FROM olist_customers_dataset  
WHERE customer_state = 'SP';
```

3.

SelectQuery.sql\* SQL 2\*

```
1 -- SELECT customer_id, customer_city, customer_state
2   -- FROM olist_customers_dataset
3   -- WHERE customer_state = 'SP';
4   SELECT order_id, payment_type, payment_value
5     FROM olist_order_payments_dataset
6    WHERE payment_value > 100;
7
8
9
```

	order_id	payment_type	payment_value
1	ba78997921bbcd1373bb41e913ab953	credit_card	107.78
2	42fdf880ba16b47b59251dd489d441a	credit_card	128.45
3	1f78449c87a54faf9e96e88ba1491fa9	credit_card	341.09
4	d88e0d5fa41661ce03cf6cf336527646	credit_card	188.73
5	2480f727e869fdeb397244a21b721b67	credit_card	141.9
6	cf95215a722f3ebf29e6bbab87a29e61	credit_card	102.66
7	769214176682788a92801d8907falb40	credit_card	105.28
8	12e5cfe0e4716b59af80e0f4a3bd6570	credit_card	157.45
9	61059985a6fc0ad64e95d9944caacdad	credit_card	132.04
10	8ac09207f415d55acff302df7d6a895c	credit_card	244.15

Execution finished without errors.  
Result: 51776 rows returned in 359ms  
At line 1:  
-- SELECT customer\_id, customer\_city, customer\_state

4.

SelectQuery.sql\* SelectWhere.sql SQL 3\*

```
1   SELECT customer_id, customer_city, customer_state
2     FROM olist_customers_dataset
3    WHERE customer_state = 'SP'
4    ORDER BY customer_city ASC;
5
```

	customer_id	customer_city	customer_state
1	ed35271185aa3ecf2e421970e47e473f	adamantina	SP
2	9c8238260ea546e75c56fc27c0fedfc8	adamantina	SP
3	f5219627819a30474a6ef14e1598e7f4	adamantina	SP
4	5684f8bf0ee27f2003ba2380c4ba8b08	adamantina	SP
5	cc6a744bal4a060493fb64b97087006c	adamantina	SP
6	9872961494c04c3939cc4fa43bfaeb75	adamantina	SP
7	4d32a98e390433alld785b4078826807	adamantina	SP
8	370e9b111968bd747c429eb56c54c8e3	adamantina	SP
9	a54c7810d4a0a3563f11fb6ce6acf771	adamantina	SP
10	8d530bb376a32cla80884c941bfa3789	adamantina	SP

Execution finished without errors.  
Result: 41746 rows returned in 450ms  
At line 1:  
SELECT customer\_id, customer\_city, customer\_state

5.

The screenshot shows a SQL IDE interface with three tabs: 'SelectQuery.sql\*', 'SelectWhere.sql', and 'SQL 3\*'. The 'SQL 3\*' tab contains the following SQL code:

```
3 -- WHERE customer_state = 'SP'
4 -- ORDER BY customer_city ASC;
5
6 SELECT order_id, payment_type, payment_value
7 FROM olist_order_payments_dataset
8 WHERE payment_value > 100
9 ORDER BY payment_value DESC;
10
11
```

Below the code, the results are displayed in a table:

	order_id	payment_type	payment_value
1	03caa2c082116e1d31e67e9ae3700499	credit_card	13664.08
2	736e1922ae60d0d6a89247b851902527	boleto	7274.88
3	0812eb902a67711a1cb742b3cdAA65ae	credit_card	6929.31
4	fefacc66af859508bf1a7934eable97f	boleto	6922.21
5	f5136e38d1a14a4dbd87dff67da82701	boleto	6726.66
6	2cc9089445046817a7539d90805e6e5a	boleto	6081.54
7	a96610ab360d42a2e5335a3998b4718a	credit_card	4950.34
8	b4c4b76c642808cbe472a32b86cccc95	credit_card	4809.44
9	199af31afc78c699f0dbf71fb178d4d4	credit_card	4764.34
10	8dbc85d1447242f3b127ddaa390d56e19	credit_card	4681.78

Execution finished without errors.  
Result: 51776 rows returned in 761ms  
At line 1:  
-- SELECT customer\_id, customer\_city, customer\_state

6.

The screenshot shows a SQL IDE interface with three tabs: 'SelectQuery.sql\*', 'SelectWhere.sql', and 'SQL 3\*'. The 'SQL 3\*' tab contains the following SQL code:

```
8 -- WHERE payment_value > 100
9 -- ORDER BY payment_value DESC;
10
11 SELECT product_id, product_category_name
12 FROM olist_products_dataset
13 WHERE product_category_name IS NOT NULL
14 ORDER BY product_category_name ASC;
15
16
```

Below the code, the results are displayed in a table:

	product_id	product_category_name
1	07f01b6fcacclbl87a71e5074199db2d	agro_industria_e_comercio
2	613d093272cb8f74f25a01e430155a6a	agro_industria_e_comercio
3	980ecbcc15fe174ec1e5757c4d75b1bf	agro_industria_e_comercio
4	bald7e7eelf055d252a2faa8ea3cea9b	agro_industria_e_comercio
5	137ace556a03792cdc43f91ec621426d	agro_industria_e_comercio
6	eble85561435be5d1760939dac8c881c	agro_industria_e_comercio
7	b5015f9057f27b5bdefde4ca99368b54	agro_industria_e_comercio
8	51d1db0b0ed8fee9994a5bae549a6160	agro_industria_e_comercio
9	0b2a1288e8ba64c797e7586c8df75602	agro_industria_e_comercio
10	3e3f442db862cb6fe99389a41b7acb84	agro_industria_e_comercio

Execution finished without errors.  
Result: 32341 rows returned in 282ms  
At line 1:  
-- SELECT customer\_id, customer\_city, customer\_state

7.

The screenshot shows a SQL development environment with three tabs at the top: 'SelectQuery.sql\*', 'SelectWhere.sql', and 'SQL 3\*'. The 'SelectQuery.sql\*' tab contains the following SQL code:

```
8    -- WHERE payment_value > 100
9    -- ORDER BY payment_value DESC;
10
11   SELECT product_id, product_category_name
12     FROM olist_products_dataset
13    WHERE product_category_name IS NOT NULL
14    ORDER BY product_category_name ASC;
15
16
```

Below the code is a results grid with two columns: 'product\_id' and 'product\_category\_name'. The data is as follows:

	product_id	product_category_name
1	07f01b6fcacclbl87a7le5074199db2d	agro_industria_e_comercio
2	613d093272cb8f74f25a0le430155a6a	agro_industria_e_comercio
3	980ecbcc15fe174ecle5757c4d75b1bf	agro_industria_e_comercio
4	bald7e7eelf055d252a2faa8ea3cea9b	agro_industria_e_comercio
5	137ace556a03792cdc43f91ec621426d	agro_industria_e_comercio
6	eble85561435be5d1760939dac8c881c	agro_industria_e_comercio
7	b5015f057f27b5bdefde4ca99368b54	agro_industria_e_comercio
8	51d1db0b0ed8fee9994a5bae549a6160	agro_industria_e_comercio
9	0b2a1288e8ba64c797e7586c8df75602	agro_industria_e_comercio
10	3e3f442db862cb6fe99389a41b7acb84	agro_industria_e_comercio

At the bottom of the interface, there is a message: 'Execution finished without errors. Result: 32341 rows returned in 282ms'.

8.

The screenshot shows a SQL development environment with four tabs at the top: 'SelectQuery.sql\*', 'SelectWhere.sql', 'SelectOrder.sql', and 'SQL 4\*'. The 'SelectQuery.sql\*' tab contains the following SQL code:

```
1   SELECT customer_state, COUNT(customer_id) AS total_customers
2     FROM olist_customers_dataset
3   GROUP BY customer_state;
4
```

Below the code is a results grid with two columns: 'customer\_state' and 'total\_customers'. The data is as follows:

	customer_state	total_customers
1	AC	81
2	AL	413
3	AM	148
4	AP	68
5	BA	3380
6	CE	1336
7	DF	2140
8	ES	2033
9	GO	2020
10	MA	747

At the bottom of the interface, there is a message: 'Execution finished without errors. Result: 27 rows returned in 206ms'.

9.

The screenshot shows a SQL IDE interface with two tabs open: "SelectQuery.sql" and "SQL 4\*". The "SelectQuery.sql" tab contains the following code:

```
1 SELECT customer_state, COUNT(customer_id) AS total_customers
2 FROM olist_customers_dataset
3 GROUP BY customer_state;
4
5 SELECT payment_type, SUM(payment_value) AS total_payment
6 FROM olist_order_payments_dataset
7 GROUP BY payment_type;
8
```

The results of the second query are displayed in a table:

payment_type	total_payment
boleto	2869361.27
credit_card	12542084.19
debit_card	217989.79
not_defined	0.0
voucher	379436.87

The "SQL 4\*" tab shows the execution message:

```
Execution finished without errors.
Result: 5 rows returned in 212ms
At line 5:
SELECT payment_type, SUM(payment_value) AS total_payment
```

10.

The screenshot shows a SQL IDE interface with two tabs open: "SelectGroupBy.sql" and "SQL 5\*". The "SelectGroupBy.sql" tab contains the following code:

```
1 SELECT
2     oi.order_id,
3     oi.product_id,
4     p.product_category_name,
5     oi.price
6 FROM olist_order_items_dataset oi
7 INNER JOIN olist_products_dataset p
8     ON oi.product_id = p.product_id;
```

The results of the query are displayed in a table:

	order_id	product_id	product_category_name
1	00010242fe8c5a6dlba2dd792cb16214	4244733e06e7ecb4970a6e2683c13e61	cool_stuff
2	00018f77f2f0320c557190d7a144bdd3	e5f2d52b802189ee658865ca93d83a8f	pet_shop
3	000229ec398224ef6ca0657da4fc703e	c777355d18b72b67abbeef9df44fd0fd	moveis_decoracao
4	00024acbcdf0a6daale931b038114c75	7634dal52a4610f1595efa32f14722fc	perfumaria
5	00042b26cf59d7ce69dfabb4e55b4fd9	ac6c3623068f30de03045865e4e10089	ferramentas_jardim
6	00048cc3ae777c65dbb7d2a0634bclea	ef92defde845ab8450f9d70c526ef70f	utilidades_domesticas
7	00054e8431b9d7675808bcb819fb4a32	8d4f2bb7e93e6710a28f34fa83ee7d28	telefonia
8	000576fe39319847cbb9d288c5617fa6	557d850972a7d6f792fd18ae1400d9b6	ferramentas_jardim
9	0005ala1728c9d785b8e2b08b904576c	310ae3c140ff94b03219ad0adc3c778f	beleza_saude
...	...	...	...

The "SQL 5\*" tab shows the execution message:

```
Execution finished without errors.
Result: 112650 rows returned in 1189ms
At line 1:
SELECT
```

11.

The screenshot shows a SQL IDE interface with a toolbar at the top and several tabs open. The active tab is titled "SQL 6\*". Below the tabs, there is a code editor containing the following SQL query:

```
1 SELECT
2     p.product_id,
3     p.product_category_name,
4     oi.order_id
5 FROM olist_products_dataset p
6 LEFT JOIN olist_order_items_dataset oi
7     ON p.product_id = oi.product_id;
```

Below the code editor is a results grid displaying the output of the query. The columns are labeled "product\_id", "product\_category\_name", and "order\_id". The data is as follows:

	product_id	product_category_name	order_id
1	le9e8ef04dbcff4541ed26657ea517e5	perfumaria	e17e4f88e31525f7deef66779844ddce
2	3aa071139cb16b67ca9e5dea641aaa2f	artes	5236307716393b7114b53ee991f36956
3	96bd76ec8810374edlb65e291975717f	esporte_lazer	01f66e58769f84129811d43eef187fb
4	cef67bcfe19066a932b7673e239eb23d	bebés	143d00a4f2dde4e0364ee1821577adb3
5	9dc1a7de274444849c219cff195d0b71	utilidades_domesticas	86cafb8794cb99a9b1b77fc8e48fbffff
6	41d3672d4792049fa1779bb35283ed13	instrumentos_musicais	c214058828b43a44f352b56ff2d5c0a5
7	732bd381ad09e530fe0a5f457d81becb	cool_stuff	9632facd8bd95315d63a23bf616d85b0
8	732bd381ad09e530fe0a5f457d81becb	cool_stuff	c6343db6c1801f9c3301166f02931116
9	2548af3e6e77a690cf3eb6368e9ab61e	moveis_decoracao	6ecf7023e8dd4ec8b08746c35b9fcb60
10	2548af3e6e77a690cf3eb6368e9ab61e	moveis_decoracao	6ecf7023e8dd4ec8b08746c35b9fcb60

At the bottom of the results grid, the following message is displayed:

```
Execution finished without errors.
Result: 112650 rows returned in 3967ms
At line 1:
SELECT
```

The screenshot shows a SQL development environment with three main panes. The top pane contains a SQL editor window titled "SQL 1\*" with the following query:

```
1  SELECT order_id, payment_value
2  FROM olist_order_payments_dataset
3  WHERE payment_value = (
4      SELECT MAX(payment_value)
5      FROM olist_order_payments_dataset
6  );
7
```

The middle pane displays the results of the query as a table:

	order_id	payment_value
1	03caa2c082116e1d31e67e9ae3700499	13664.08

The bottom pane shows the execution details:

```
Execution finished without errors.
Result: 1 rows returned in 117ms
At line 1:
SELECT order_id, payment_value
FROM olist_order_payments_dataset
WHERE payment_value = (
```

12.

13.

Subquery.sql SQL 2\*

```
1 SELECT
2     SUM(payment_value) AS total_revenue
3 FROM olist_order_payments_dataset;
4
```

total_revenue
1 16008872.12

Execution finished without errors.  
Result: 1 rows returned in 53ms  
At line 1:  
SELECT  
 SUM(payment\_value) AS total\_revenue  
FROM olist\_order\_payments\_dataset;

14.

```
1 SELECT
2     AVG(payment_value) AS avg_payment_value
3 FROM olist_order_payments_dataset;
```

avg_payment_value
154.100380416996

Execution finished without errors.  
Result: 1 rows returned in 58ms  
At line 6:  
SELECT  
 AVG(payment\_value) AS avg\_payment\_value  
FROM olist\_order\_payments\_dataset;

15.

```
9
10    SELECT
11        payment_type,
12        SUM(payment_value) AS total_payment
13    FROM olist_order_payments_dataset
14    GROUP BY payment_type;
15
```

	payment_type	total_payment
1	boleto	2869361.27
2	credit_card	12542084.19
3	debit_card	217989.79
4	not_defined	0.0
5	voucher	379436.87

Execution finished without errors.

Result: 5 rows returned in 229ms

At line 10:

```
SELECT
    payment_type,
    SUM(payment_value) AS total_payment
FROM olist_order_payments_dataset
GROUP BY payment_type;
```

16.

```
Subquery.sql SQL AggregateFunctions.sql SQL 3*
```

```
1 CREATE VIEW vw_product_sales AS
2     SELECT
3         product_id,
4         SUM(price) AS total_product_sales
5     FROM olist_order_items_dataset
6     GROUP BY product_id;
7
8     SELECT * FROM vw_product_sales;
9
10
```

	product_id	total_product_sales
1	00066f42aeeb9f3007548bb9d3f33c38	101.65
2	00088930e925c41fd95ebfe695fd2655	129.9
3	0009406fd7479715e4bef61dd91f2462	229.0
4	000b8f95fc9e0096488278317764d19	117.8
5	000d9be29b5207b54e86aalblac54872	199.0
6	0011c512eb256aa0dbbb544d8dffcf6e	52.0
7	00126f27a813e03687a6ca496d1009d01	499.0

Execution finished without errors.

Result: 32951 rows returned in 493ms

At line 8:

```
SELECT * FROM vw_product_sales;
```

17.

The screenshot shows a SQL development environment with multiple tabs at the top: Subquery.sql, AggregateFunctions.sql, Views.sql, and SQL 4\*. The SQL 4\* tab is active and contains the following code:

```
3 GROUP BY customer_unique_id;
4
5
6 CREATE INDEX idx_customers_unique_id
7 ON olist_customers_dataset (customer_unique_id);
8
9
10 SELECT customer_unique_id, COUNT(customer_id)
11 FROM olist_customers_dataset
12 GROUP BY customer_unique_id;
```

Below the code is a results grid showing the output of the query:

	customer_unique_id	COUNT(customer_id)
1	0000366f3b9a7992bf8c76cfdf3221e2	1
2	0000b849f77a49e4a4ce2b2a4ca5be3f	1
3	0000f46a3911fa3c0805444483337064	1
4	0000f6ccb0745a6a4b88665a16c9f078	1
5	0004aac84e0df4da2b147fca70cf8255	1
6	0004bd2a26a76fe21f786e4fb080607f	1
7	00050ab1314a0a55a6ea13af7191faef	1

At the bottom of the results grid, the message "Execution finished without errors." is displayed, followed by "Result: 96096 rows returned in 2175ms". Below this, the original query is repeated:

```
SELECT customer_unique_id, COUNT(customer_id)
FROM olist_customers_dataset
GROUP BY customer_unique_id;
```