



```

1 • use nayan;
2 • CREATE TABLE Sale (
3     Product VARCHAR(50),
4     Category VARCHAR(50),
5     Quantity INT,
6     Price DECIMAL(10, 2),
7     Date DATETIME
8 );
9
10 • INSERT INTO Sale (Product, Category, Quantity, Price, Date) VALUES
11 ('Headphones', 'Appliances', 45, 475.29, '2023-09-23 00:00:00'),
12 ('Headphones', 'Accessories', 48, 334.64, '2023-01-28 00:00:00'),
13 ('Keyboard', 'Furniture', 65, 16.65, '2023-07-19 00:00:00'),
14 ('Mouse', 'Accessories', 68, 315.19, '2023-03-03 00:00:00'),
15 ('Camera', 'Accessories', 68, 340.09, '2023-12-08 00:00:00'),
16 ('Smartphone', 'Furniture', 10, 486.25, '2023-12-20 00:00:00'),
17 ...

```

Output

📄 Action Output ▾

	#	Time	Action	Message
✓	47	15:24:40	SELECT Product, COUNT(*) AS Count FROM Sale WHERE Price = (SELECT MIN(Price) F...	1 row(s) returned
✓	48	15:29:11	use nayan	0 row(s) affected
✗	49	15:29:25	CREATE TABLE Sale ( Product VARCHAR(50), Category VARCHAR(50), Quantity I...	Error Code: 1050. Table 'sale' already exists
✓	50	15:29:37	INSERT INTO Sale (Product, Category, Quantity, Price, Date) VALUES ('Headphones', 'Appl...	100 row(s) affected Records: 100 Duplicates: 0 Warnings: 0



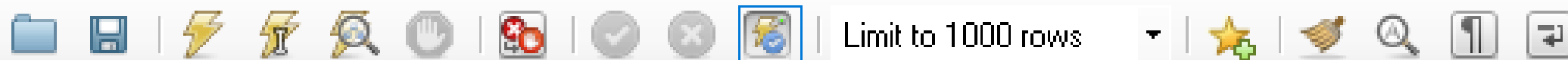
Limit to 1000 rows



```
16 ('Smartphone', 'Furniture', 10, 486.25, '2023-12-20 00:00:00'),
17 ('Mouse', 'Clothing', 84, 440.31, '2023-02-14 00:00:00'),
18 ('Camera', 'Clothing', 22, 259.72, '2023-10-18 00:00:00'),
19 ('Speaker', 'Appliances', 37, 37.3, '2023-03-30 00:00:00'),
20 ('Keyboard', 'Appliances', 88, 231.07, '2023-02-03 00:00:00'),
21 ('Speaker', 'Accessories', 71, 19.79, '2023-05-14 00:00:00'),
22 ('Laptop', 'Furniture', 89, 226.44, '2023-08-21 00:00:00'),
23 ('Headphones', 'Furniture', 89, 490, '2023-09-13 00:00:00'),
24 ('Headphones', 'Electronics', 13, 186.13, '2023-02-06 00:00:00'),
25 ('Printer', 'Electronics', 59, 245.64, '2023-09-14 00:00:00'),
26 ('Tablet', 'Accessories', 66, 347.44, '2023-10-18 00:00:00'),
27 ('Camera', 'Appliances', 40, 441.43, '2023-07-17 00:00:00'),
28 ('Tablet', 'Clothing', 88, 459.94, '2023-09-12 00:00:00'),
29 ('Headphones', 'Clothing', 47, 116.24, '2023-03-22 00:00:00'),
30 ('Camera', 'Accessories', 89, 286.94, '2023-05-17 00:00:00'),
31 ('Keyboard', 'Furniture', 82, 433.9, '2023-07-09 00:00:00'),
```

--

```
34 ('Laptop', 'Furniture', 78, 461.37, '2023-10-19 00:00:00'),
35 ('Monitor', 'Appliances', 73, 50.73, '2023-12-14 00:00:00'),
36 ('Camera', 'Appliances', 10, 146.08, '2023-06-18 00:00:00'),
37 ('Smartphone', 'Accessories', 21, 14.58, '2023-10-20 00:00:00'),
38 ('Smartphone', 'Electronics', 81, 422.75, '2023-06-26 00:00:00'),
39 ('Headphones', 'Electronics', 70, 327.12, '2023-01-26 00:00:00'),
40 ('Laptop', 'Clothing', 80, 422.28, '2023-11-20 00:00:00'),
41 ('Printer', 'Furniture', 48, 139.72, '2023-12-26 00:00:00'),
42 ('Printer', 'Furniture', 65, 204.93, '2023-10-19 00:00:00'),
43 ('Monitor', 'Electronics', 83, 280.88, '2023-04-25 00:00:00'),
44 ('Monitor', 'Appliances', 89, 90.82, '2023-10-14 00:00:00'),
45 ('Keyboard', 'Clothing', 50, 191.21, '2023-01-30 00:00:00'),
46 ('Mouse', 'Clothing', 30, 81.76, '2023-08-30 00:00:00'),
47 ('Mouse', 'Appliances', 20, 289.11, '2023-10-17 00:00:00'),
48 ('Keyboard', 'Clothing', 20, 354.83, '2023-05-27 00:00:00'),
49 ('Smartphone', 'Appliances', 15, 151.35, '2023-10-01 00:00:00'),
-- ... ..
```



```
96      ('Camera', 'Appliances', 4, 422.16, '2023-03-29 00:00:00'),
97      ('Smartphone', 'Accessories', 77, 453.28, '2023-01-14 00:00:00'),
98      ('Headphones', 'Accessories', 53, 28.40, '2023-11-11 00:00:00'),
99      ('Laptop', 'Clothing', 79, 259.07, '2023-03-23 00:00:00'),
100     ('Speaker', 'Electronics', 16, 91.76, '2023-05-01 00:00:00'),
101     ('Speaker', 'Appliances', 21, 391.73, '2023-11-17 00:00:00'),
102     ('Speaker', 'Accessories', 59, 433.82, '2023-07-23 00:00:00'),
103     ('Smartphone', 'Electronics', 24, 211.58, '2023-08-09 00:00:00'),
104     ('Smartphone', 'Electronics', 80, 78.59, '2023-10-09 00:00:00'),
105     ('Mouse', 'Accessories', 14, 26.28, '2023-10-16 00:00:00'),
106     ('Speaker', 'Clothing', 86, 491.46, '2023-09-28 00:00:00'),
107     ('Monitor', 'Furniture', 49, 192.91, '2023-10-12 00:00:00'),
108     ('Printer', 'Clothing', 50, 215.84, '2023-10-04 00:00:00'),
109     ('Headphones', 'Accessories', 70, 34.79, '2023-11-21 00:00:00'),
110     ('Headphones', 'Appliances', 42, 189.09, '2023-01-23 00:00:00');
111 • select * from Sales;
```

```
111 • select * from Sales;
112 -- (***** )
113 -- 1) ( Select all columns where Quantity is greater than 20. )
114 • SELECT *
115 FROM Sale
116 WHERE Quantity > 20;
117
118 -- 2) (Retrieve Product and Price where Price is less than $30 )
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Headphones	Appliances	45	475.29	2023-09-23 00:00:00
	Headphones	Accessories	48	334.64	2023-01-28 00:00:00
	Keyboard	Furniture	65	16.65	2023-07-19 00:00:00
	Mouse	Accessories	68	315.19	2023-03-03 00:00:00
	Camera	Accessories	68	340.09	2023-12-08 00:00:00
	Smartphone	Furniture	10	486.35	2023-12-20 00:00:00

Sales 31

Output

Action Output


▼

	#	Time	Action	Message
✓	48	15:29:11	use nayan	0 row(s) affected
✗	49	15:29:25	CREATE TABLE Sale ( Product VARCHAR(50), Category VARCHAR(50), Quantity I...	Error Code: 1050. Table 'sale' already
✓	50	15:29:37	INSERT INTO Sale (Product, Category, Quantity, Price, Date) VALUES ('Headphones', 'Appl...	100 row(s) affected Records: 100 Dup
✓	51	15:34:17	select * from Sales LIMIT 0,1000	100 row(s) returned

```
117
118      -- 2) (Retrieve Product and Price where Price is less than $30.)
119 •   SELECT Product, Price
120      FROM Sale
121      WHERE Price < 30;
122
123      -- 3) Find all records where Date is in the year 2023
```

Result Grid

 Filter Rows:


Export: 

Wrap Cell Content: 

	Product	Price
▶	Keyboard	16.65
	Speaker	19.79
	Smartphone	14.58
	Monitor	26.48
	Headphones	28.40
	Mouse	26.20

Sale 32 

Output

 Action Output

▼

	#	Time	Action	Message
✖	49	15:29:25	CREATE TABLE Sale ( Product VARCHAR(50), Category VARCHAR(50), Quantity I...	Error Code: 1050. Table 'sale' already exists
✔	50	15:29:37	INSERT INTO Sale (Product, Category, Quantity, Price, Date) VALUES ('Headphones', 'Appl...	100 row(s) affected Records: 100 Duplicates: 0 Warnings: 0
✔	51	15:34:17	select * from Sales LIMIT 0, 1000	100 row(s) returned
✔	52	15:35:47	SELECT Product, Price FROM Sale \WHERE Price < 30 LIMIT 0, 1000	12 row(s) returned

```
---
123      -- 3) Find all records where Date is in the year 2023.
124  •    SELECT *
125      FROM Sale
126      WHERE YEAR(Date) = 2023;
127
128      -- 4) Select distinct Category values from the Sales table
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Headphones	Appliances	45	475.29	2023-09-23 00:00:00
	Headphones	Accessories	48	334.64	2023-01-28 00:00:00
	Keyboard	Furniture	65	16.65	2023-07-19 00:00:00
	Mouse	Accessories	68	315.19	2023-03-03 00:00:00
	Camera	Accessories	68	340.09	2023-12-08 00:00:00
	Smartphone	Furniture	10	496.25	2023-12-20 00:00:00

Sale 33

Output

Action Output

▼

	#	Time	Action	Message
✓	50	15:29:37	INSERT INTO Sale (Product, Category, Quantity, Price, Date) VALUES ('Headphones', 'Appl...	100 row(s) affected Records: 100 Duplicates: 0 Warnings: 0
✓	51	15:34:17	select * from Sales LIMIT 0, 1000	100 row(s) returned
✓	52	15:35:47	SELECT Product, Price FROM Sale WHERE Price < 30 LIMIT 0, 1000	12 row(s) returned
✓	53	15:36:24	SELECT * FROM Sale WHERE YEAR(Date) = 2023 LIMIT 0, 1000	200 row(s) returned

```
127
128 -- 4) Select distinct Category values from the Sales table.
129 • SELECT DISTINCT Category
130 FROM Sale;
131
132 -- 5) Find records where Product name contains the word 'Phone'
```

Result Grid | Filter Rows:  | Export: | Wrap Cell Content:

	Category
▶	Appliances
	Accessories
	Furniture
	Clothing
	Electronics

Sale 35 ×

Output ⋮

Action Output ▼

	#	Time	Action	Message
✓	52	15:35:47	SELECT Product, Price FROM Sale WHERE Price < 30 LIMIT 0, 1000	12 row(s) returned
✓	53	15:36:24	SELECT * FROM Sale WHERE YEAR(Date) = 2023 LIMIT 0, 1000	200 row(s) returned
✓	54	15:36:49	SELECT DISTINCT Category FROM Sale LIMIT 0, 1000	5 row(s) returned
✓	55	15:39:06	SELECT DISTINCT Category FROM Sale LIMIT 0, 1000	5 row(s) returned



```
132 -- 5) Find records where Product name contains the word 'Phone'.
133 • SELECT *
134 FROM Sales
135 WHERE Product LIKE '%Phone%';
136
137 -- 6) Select Product and Quantity where Quantity is between 10 and 50
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Headphones	Appliances	45	475.29	2023-09-23 00:00:00
	Headphones	Accessories	48	334.64	2023-01-28 00:00:00
	Smartphone	Furniture	10	486.25	2023-12-20 00:00:00
	Headphones	Furniture	89	490.00	2023-09-13 00:00:00
	Headphones	Electronics	13	186.13	2023-02-06 00:00:00
	Headphones	Clothing	47	114.24	2023-03-22 00:00:00

Sales 36

Output

Action Output

▼

	#	Time	Action	Message
✓	53	15:36:24	SELECT * FROM Sale WHERE YEAR(Date) = 2023 LIMIT 0, 1000	200 row(s) returned
✓	54	15:36:49	SELECT DISTINCT Category FROM Sale LIMIT 0, 1000	5 row(s) returned
✓	55	15:39:06	SELECT DISTINCT Category FROM Sale LIMIT 0, 1000	5 row(s) returned
✓	56	15:39:28	SELECT * FROM Sales WHERE Product LIKE '%Phone%' LIMIT 0, 1000	22 row(s) returned

```

136
137 -- 6) Select Product and Quantity where Quantity is between 10 and 50.
138 • SELECT Product, Quantity
139 FROM Sale
140 WHERE Quantity BETWEEN 10 AND 50;
141


```

Result Grid |  Filter Rows:  | Export:  | Wrap Cell Content: 

	Product	Quantity
▶	Headphones	45
	Headphones	48
	Smartphone	10
	Camera	22
	Speaker	37
	Headphones	12

Sale 37 

Output


 Action Output ▼

	#	Time	Action	Message
✓	54	15:36:49	SELECT DISTINCT Category FROM Sale LIMIT 0, 1000	5 row(s) returned
✓	55	15:39:06	SELECT DISTINCT Category FROM Sale LIMIT 0, 1000	5 row(s) returned
✓	56	15:39:28	SELECT * FROM Sales WHERE Product LIKE '%Phone%' LIMIT 0, 1000	22 row(s) returned
✓	57	15:39:51	SELECT Product, Quantity FROM Sale WHERE Quantity BETWEEN 10 AND 50 LIMIT 0, ...	94 row(s) returned

```

142 -- 7) Retrieve records where Category is either 'Electronics' or 'Furniture'.
143 • SELECT *
144 FROM Sale
145 WHERE Category IN ('Electronics', 'Furniture');
146
147 -- 8) Find records where Date is in the month of December


```

Result Grid |  Filter Rows:  | Export:  | Wrap Cell Content: 

	Product	Category	Quantity	Price	Date
►	Keyboard	Furniture	65	16.65	2023-07-19 00:00:00
	Smartphone	Furniture	10	486.25	2023-12-20 00:00:00
	Laptop	Furniture	89	226.44	2023-08-21 00:00:00
	Headphones	Furniture	89	490.00	2023-09-13 00:00:00
	Headphones	Electronics	13	186.13	2023-02-06 00:00:00
	Printer	Electronics	50	345.64	2023-06-14 00:00:00

Sale 39 ✕

Output

 Action Output ▼

	#	Time	Action	Message
✓	56	15:39:28	SELECT * FROM Sales WHERE Product LIKE '%Phone%' LIMIT 0, 1000	22 row(s) returned
✓	57	15:39:51	SELECT Product, Quantity FROM Sale WHERE Quantity BETWEEN 10 AND 50 LIMIT 0, ...	94 row(s) returned
✓	58	15:40:15	SELECT * FROM Sale WHERE Category IN ('Electronics', 'Furniture') LIMIT 0, 1000	70 row(s) returned
✓	59	15:40:24	SELECT * FROM Sale WHERE Category IN ('Electronics', 'Furniture') LIMIT 0, 1000	70 row(s) returned

```
147 -- 8) Find records where Date is in the month of December.
148 • SELECT *
149 FROM Sale
150 WHERE MONTH(Date) = 12;
151
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Camera	Accessories	68	340.09	2023-12-08 00:00:00
	Smartphone	Furniture	10	486.25	2023-12-20 00:00:00
	Monitor	Appliances	73	50.73	2023-12-14 00:00:00
	Printer	Furniture	48	139.72	2023-12-26 00:00:00
	Monitor	Electronics	33	380.49	2023-12-07 00:00:00
	Mouse	Furniture	59	323.07	2023-12-24 00:00:00

Sale 40

Output

Action Output

▼

	#	Time	Action	Message
✓	57	15:39:51	SELECT Product, Quantity FROM Sale \WHERE Quantity BETWEEN 10 AND 50 LIMIT 0, ...	94 row(s) returned
✓	58	15:40:15	SELECT * FROM Sale \WHERE Category IN ('Electronics', 'Furniture') LIMIT 0, 1000	70 row(s) returned
✓	59	15:40:24	SELECT * FROM Sale \WHERE Category IN ('Electronics', 'Furniture') LIMIT 0, 1000	70 row(s) returned
✓	60	15:40:47	SELECT * FROM Sale \WHERE MONTH(Date) = 12 LIMIT 0, 1000	20 row(s) returned

```
152      -- 9) Select records where Price is exactly $100.
153  ●   SELECT *
154      FROM Sale
155      WHERE Price = 100;
156
```

Result Grid

  Filter Rows:

Export: 





Wrap Cell Content: 

	Product	Category	Quantity	Price	Date
--	---------	----------	----------	-------	------

Sale 41 

Output

 Action Output

	#	Time	Action	Message
	58	15:40:15	SELECT * FROM Sale WHERE Category IN ('Electronics', 'Furniture') LIMIT 0, 1000	70 row(s) returned
	59	15:40:24	SELECT * FROM Sale WHERE Category IN ('Electronics', 'Furniture') LIMIT 0, 1000	70 row(s) returned
	60	15:40:47	SELECT * FROM Sale WHERE MONTH(Date) = 12 LIMIT 0, 1000	20 row(s) returned
	61	15:41:05	SELECT * FROM Sale WHERE Price = 100 LIMIT 0, 1000	0 row(s) returned

```
157      -- 10) Find all rows where Product starts with the letter 'S'.
158  •    SELECT *
159      FROM Sale
160      WHERE Product LIKE 'S%';
161
162      -- 11) Select all columns where Price is not equal to $50
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Smartphone	Furniture	10	486.25	2023-12-20 00:00:00
	Speaker	Appliances	37	37.30	2023-03-30 00:00:00
	Speaker	Accessories	71	19.79	2023-05-14 00:00:00
	Smartphone	Accessories	21	14.58	2023-10-20 00:00:00
	Smartphone	Electronics	81	422.75	2023-06-26 00:00:00
	Smartphone	Appliances	15	151.35	2023-10-01 00:00:00

Sale 42

Output


Action Output

	#	Time	Action	Message
✓	59	15:40:24	SELECT * FROM Sale WHERE Category IN ('Electronics', 'Furniture') LIMIT 0, 1000	70 row(s) returned
✓	60	15:40:47	SELECT * FROM Sale WHERE MONTH(Date) = 12 LIMIT 0, 1000	20 row(s) returned
✓	61	15:41:05	SELECT * FROM Sale WHERE Price = 100 LIMIT 0, 1000	0 row(s) returned
✓	62	15:41:38	SELECT * FROM Sale WHERE Product LIKE 'S%' LIMIT 0, 1000	42 row(s) returned

```

162      -- 11) Select all columns where Price is not equal to $50.
163  •   SELECT *
164      FROM Sale
165      WHERE Price <> 50;
166
167      -- 12) Retrieve records where Quantity is greater than 10 and Price is less than $50


```

Result Grid |  Filter Rows:  | Export:  | Wrap Cell Content: 

	Product	Category	Quantity	Price	Date
▶	Headphones	Appliances	45	475.29	2023-09-23 00:00:00
	Headphones	Accessories	48	334.64	2023-01-28 00:00:00
	Keyboard	Furniture	65	16.65	2023-07-19 00:00:00
	Mouse	Accessories	68	315.19	2023-03-03 00:00:00
	Camera	Accessories	68	340.09	2023-12-08 00:00:00
	Smartphone	Furniture	10	196.25	2023-12-20 00:00:00

Sale 43 

Output 

 Action Output 

	#	Time	Action	Message
✓	60	15:40:47	SELECT * FROM Sale WHERE MONTH(Date) = 12 LIMIT 0, 1000	20 row(s) returned
✓	61	15:41:05	SELECT * FROM Sale WHERE Price = 100 LIMIT 0, 1000	0 row(s) returned
✓	62	15:41:38	SELECT * FROM Sale WHERE Product LIKE 'S%' LIMIT 0, 1000	42 row(s) returned
✓	63	15:42:27	SELECT * FROM Sale WHERE Price <> 50 LIMIT 0, 1000	200 row(s) returned

166

167 -- 12) Retrieve records where Quantity is greater than 10 and Price is less than \$50.

168 • **SELECT \***

169 **FROM** Sale

170 **WHERE** Quantity > 10 **AND** Price < 50;


171

**Result Grid** |   Filter Rows:  | Export:  | Wrap Cell Content: 

	Product	Category	Quantity	Price	Date
▶	Keyboard	Furniture	65	16.65	2023-07-19 00:00:00
	Speaker	Appliances	37	37.30	2023-03-30 00:00:00
	Speaker	Accessories	71	19.79	2023-05-14 00:00:00
	Smartphone	Accessories	21	14.58	2023-10-20 00:00:00
	Monitor	Accessories	83	26.48	2023-01-25 00:00:00
	Headphones	Accessories	59	28.40	2023-11-11 00:00:00

Sale 44 ✕

Output

 Action Output ▼

	#	Time	Action	Message
✓	61	15:41:05	SELECT * FROM Sale WHERE Price = 100 LIMIT 0, 1000	0 row(s) returned
✓	62	15:41:38	SELECT * FROM Sale WHERE Product LIKE 'S%' LIMIT 0, 1000	42 row(s) returned
✓	63	15:42:27	SELECT * FROM Sale WHERE Price <> 50 LIMIT 0, 1000	200 row(s) returned
✓	64	15:42:43	SELECT * FROM Sale WHERE Quantity > 10 AND Price < 50 LIMIT 0, 1000	16 row(s) returned



```
172      -- 13) Find all rows where Date is either before or after 2023.
173  •  SELECT *
174      FROM Sale
175      WHERE YEAR(Date) <> 2023;
176
```

Result Grid

  Filter Rows:


Export: 

Wrap Cell Content: 


	Product	Category	Quantity	Price	Date
--	---------	----------	----------	-------	------





Sale 45 

Output



Action Output



	#	Time	Action	Message
	62	15:41:38	SELECT * FROM Sale WHERE Product LIKE 'S%' LIMIT 0,1000	42 row(s) returned
	63	15:42:27	SELECT * FROM Sale WHERE Price <> 50 LIMIT 0,1000	200 row(s) returned
	64	15:42:43	SELECT * FROM Sale WHERE Quantity > 10 AND Price < 50 LIMIT 0,1000	16 row(s) returned
	65	15:43:08	SELECT * FROM Sale WHERE YEAR(Date) <> 2023 LIMIT 0,1000	0 row(s) returned

```
176
177      -- 14) Select Product and Price where Price is between $20 and $80 inclusive.
178 •   SELECT Product, Price
179     FROM Sale
180    WHERE Price BETWEEN 20 AND 80;
181
```

Result Grid | Filter Rows:  | Export: | Wrap Cell Content:

	Product	Price
▶	Speaker	37.30
	Monitor	50.73
	Speaker	66.94
	Headphones	60.40
	Camera	55.26
	Monitor	26.10

Sale 46 x

Output

Action Output ▼

	#	Time	Action	Message
✓	63	15:42:27	SELECT * FROM Sale WHERE Price <> 50 LIMIT 0, 1000	200 row(s) returned
✓	64	15:42:43	SELECT * FROM Sale WHERE Quantity > 10 AND Price < 50 LIMIT 0, 1000	16 row(s) returned
✓	65	15:43:08	SELECT * FROM Sale WHERE YEAR(Date) <> 2023 LIMIT 0, 1000	0 row(s) returned
✓	66	15:43:34	SELECT Product, Price FROM Sale WHERE Price BETWEEN 20 AND 80 LIMIT 0, 1000	26 row(s) returned

181

182 -- 15) Find records where Quantity is either 5 or 15.

183 • SELECT \*

184 FROM Sale

185 WHERE Quantity IN (5, 15);

186

187 -- 16) Select all columns where Product contains 'Headphones' or 'Sneaker'

Result Grid |  Filter Rows:  | Export:  | Wrap Cell Content: 

	Product	Category	Quantity	Price	Date
	Smartphone	Appliances	15	151.35	2023-10-01 00:00:00
	Printer	Furniture	5	242.69	2023-07-21 00:00:00
	Printer	Appliances	15	118.42	2023-07-15 00:00:00
	Smartphone	Appliances	15	151.35	2023-10-01 00:00:00
	Printer	Furniture	5	242.69	2023-07-21 00:00:00
	Printer	Appliances	15	118.42	2023-07-15 00:00:00

ale 47 x

Output

 Action Output ▾

#	Time	Action	Message
64	15:42:43	SELECT * FROM Sale WHERE Quantity > 10 AND Price < 50 LIMIT 0, 1000	16 row(s) returned
65	15:43:08	SELECT * FROM Sale WHERE YEAR(Date) <> 2023 LIMIT 0, 1000	0 row(s) returned
66	15:43:34	SELECT Product, Price FROM Sale WHERE Price BETWEEN 20 AND 80 LIMIT 0, 1000	26 row(s) returned
67	15:44:23	SELECT * FROM Sale WHERE Quantity IN (5, 15) LIMIT 0, 1000	6 row(s) returned

```
186
187 -- 16) Select all columns where Product contains 'Headphones' or 'Speaker'.
188 • SELECT *
189 FROM Sale
190 WHERE Product LIKE '%Headphones%' OR Product LIKE '%Speaker%';
191
192 -- 17) Retrieve records where Category is neither 'Clothing' nor 'Accessories'
```

Result Grid

  Filter Rows:

Export: 

Wrap Cell Content: 

	Product	Category	Quantity	Price	Date
▶	Headphones	Appliances	45	475.29	2023-09-23 00:00:00
	Headphones	Accessories	48	334.64	2023-01-28 00:00:00
	Speaker	Appliances	37	37.30	2023-03-30 00:00:00
	Speaker	Accessories	71	19.79	2023-05-14 00:00:00
	Headphones	Furniture	89	490.00	2023-09-13 00:00:00
	Headphones	Electronics	12	186.12	2023-02-06 00:00:00

Sale 48 

Output

 Action Output 

	#	Time	Action	Message
✓	65	15:43:08	SELECT * FROM Sale WHERE YEAR(Date) <> 2023 LIMIT 0, 1000	0 row(s) returned
✓	66	15:43:34	SELECT Product, Price FROM Sale WHERE Price BETWEEN 20 AND 80 LIMIT 0, 1000	26 row(s) returned
✓	67	15:44:23	SELECT * FROM Sale WHERE Quantity IN (5, 15) LIMIT 0, 1000	6 row(s) returned
✓	68	15:45:01	SELECT * FROM Sale WHERE Product LIKE '%Headphones%' OR Product LIKE '%Speak...	50 row(s) returned

```
192      -- 17) Retrieve records where Category is neither 'Clothing' nor 'Accessories'.
193  ●    SELECT *
194      FROM Sale
195      WHERE Category NOT IN ('Clothing', 'Accessories');
196
197      -- 18) Find all records where Price is in the top 5 highest prices
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Headphones	Appliances	45	475.29	2023-09-23 00:00:00
	Keyboard	Furniture	65	16.65	2023-07-19 00:00:00
	Smartphone	Furniture	10	486.25	2023-12-20 00:00:00
	Speaker	Appliances	37	37.30	2023-03-30 00:00:00
	Keyboard	Appliances	88	231.07	2023-02-03 00:00:00
	Laptop	Furniture	80	226.44	2023-08-21 00:00:00

Sale 49

Output



Action Output


▼


	#	Time	Action	Message
✓	66	15:43:34	SELECT Product, Price FROM Sale WHERE Price BETWEEN 20 AND 80 LIMIT 0, 1000	26 row(s) returned
✓	67	15:44:23	SELECT * FROM Sale WHERE Quantity IN (5, 15) LIMIT 0, 1000	6 row(s) returned
✓	68	15:45:01	SELECT * FROM Sale WHERE Product LIKE '%Headphones%' OR Product LIKE '%Speak...	50 row(s) returned
✓	69	15:47:46	SELECT * FROM Sale WHERE Category NOT IN ('Clothing', 'Accessories') LIMIT 0, 1000	108 row(s) returned

```
204 -- 19) Select Product and Date where Date is in the last 30 days.
205 • SELECT Product, Date
206 FROM Sale
207 WHERE Date >= CURDATE() - INTERVAL 30 DAY;
208
209 -- 20) Retrieve records where Quantity is less than or equal to 10 and Price is greater
```


Result Grid

  Filter Rows:

Export: 





Wrap Cell Content: 

	Product	Date
--	---------	------

Sale 51 

Output

 Action Output 

	#	Time	Action	Message
	71	15:48:16	SELECT * FROM Sale WHERE Price IN (SELECT DISTINCT Price FROM Sales ORDER B...	Error Code: 1235. This ve
	72	15:48:25	SELECT * FROM Sale WHERE Price IN (SELECT DISTINCT Price FROM Sales ORDER B...	Error Code: 1235. This ve
	73	15:48:58	SELECT Product, Date FROM Sale WHERE Date >= CURDATE() - INTERVAL 30 DAY LI...	0 row(s) returned
	74	15:49:06	SELECT Product, Date FROM Sale WHERE Date >= CURDATE() - INTERVAL 30 DAY LI...	0 row(s) returned

```
209      -- 20) Retrieve records where Quantity is less than or equal to 10 and Price is greater than $30.
210  •    SELECT *
211      FROM Sale
212      WHERE Quantity <= 10 AND Price > 30;
213
214      -- 21) Select all records from the Sales table and order them by Price in descending order.
215  •    SELECT *
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Smartphone	Furniture	10	486.25	2023-12-20 00:00:00
	Camera	Appliances	10	146.08	2023-06-18 00:00:00
	Printer	Clothing	10	449.06	2023-03-11 00:00:00
	Tablet	Clothing	1	163.48	2023-07-18 00:00:00
	Keyboard	Accessories	1	483.05	2023-10-02 00:00:00
	Speaker	Clothing	6	271.27	2023-03-25 00:00:00

Sale 52

Output

Action Output

▼

	#	Time	Action	Message
✖	72	15:48:25	SELECT * FROM Sale WHERE Price IN (SELECT DISTINCT Price FROM Sales ORDER B...	Error Code: 1235. This version of MySQL doe
✔	73	15:48:58	SELECT Product, Date FROM Sale WHERE Date >= CURDATE() - INTERVAL 30 DAY LI...	0 row(s) returned
✔	74	15:49:06	SELECT Product, Date FROM Sale WHERE Date >= CURDATE() - INTERVAL 30 DAY LI...	0 row(s) returned
✔	75	15:49:25	SELECT * FROM Sale WHERE Quantity <= 10 AND Price > 30 LIMIT 0, 1000	22 row(s) returned

```
213
214 -- 21) Select all records from the Sales table and order them by Price in descending order.
215 • SELECT *
216 FROM Sale
217 ORDER BY Price DESC;
218
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Category	Quantity	Price	Date
▶	Monitor	Furniture	43	492.18	2023-12-14 00:00:00
	Monitor	Furniture	43	492.18	2023-12-14 00:00:00
	Speaker	Clothing	86	491.46	2023-09-28 00:00:00
	Speaker	Clothing	86	491.46	2023-09-28 00:00:00
	Headphones	Furniture	89	490.00	2023-09-13 00:00:00
	Headphones	Furniture	89	490.00	2023-09-13 00:00:00

Sale 53 x

Output

Action Output


▼

	#	Time	Action	Message
✓	73	15:48:58	SELECT Product, Date FROM Sale WHERE Date >= CURDATE() - INTERVAL 30 DAY LI...	0 row(s) returned
✓	74	15:49:06	SELECT Product, Date FROM Sale WHERE Date >= CURDATE() - INTERVAL 30 DAY LI...	0 row(s) returned
✓	75	15:49:25	SELECT * FROM Sale WHERE Quantity <= 10 AND Price > 30 LIMIT 0, 1000	22 row(s) returned
✓	76	15:49:49	SELECT * FROM Sale ORDER BY Price DESC LIMIT 0, 1000	200 row(s) returned



```
219 -- 22) Select Product and Quantity, and group the results by Product, showing the total quantity for each product.
220 • SELECT Product, SUM(Quantity) AS TotalQuantity
221 FROM Sale
222 GROUP BY Product;
223
224 -- 23) Select the average Price of all products in the Sales table.
225 • SELECT AVG(Price) AS AveragePrice
```

Result Grid

  Filter Rows:


Export: 

Wrap Cell Content: 

	Product	TotalQuantity
▶	Headphones	1382
	Keyboard	906
	Mouse	1042
	Camera	602
	Smartphone	844
	Speaker	650

Result 54 

Output

 Action Output

▼

	#	Time	Action	Message
✓	74	15:49:06	SELECT Product, Date FROM Sale WHERE Date >= CURDATE() - INTERVAL 30 DAY LI...	0 row(s) returned
✓	75	15:49:25	SELECT * FROM Sale WHERE Quantity <= 10 AND Price > 30 LIMIT 0, 1000	22 row(s) returned
✓	76	15:49:49	SELECT * FROM Sale ORDER BY Price DESC LIMIT 0, 1000	200 row(s) returned
✓	77	15:50:16	SELECT Product, SUM(Quantity) AS TotalQuantity FROM Sale GROUP BY Product LIMIT 0...	10 row(s) returned

```
223
224 -- 23) Select the average Price of all products in the Sales table.
225 • SELECT AVG(Price) AS AveragePrice
226 FROM Sale;
227
228 -- 24)Select Category and the total Quantity sold for each category.
229 • SELECT Category, sum (Quantity) AS TotalQuantity
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	AveragePrice
▶	256.125700

Result 55

Output

Action Output

▼

	#	Time	Action	Message
✓	75	15:49:25	SELECT * FROM Sale WHERE Quantity <= 10 AND Price > 30 LIMIT 0,1000	22 row(s) returned
✓	76	15:49:49	SELECT * FROM Sale ORDER BY Price DESC LIMIT 0,1000	200 row(s) returned
✓	77	15:50:16	SELECT Product, SUM(Quantity) AS TotalQuantity FROM Sale GROUP BY Product LIMIT 0...	10 row(s) returned
✓	78	15:50:49	SELECT AVG(Price) AS AveragePrice FROM Sale LIMIT 0,1000	1 row(s) returned

```
233
234 -- 25) Find the maximum Price for each Category.
235 • SELECT Category, MAX(Price) AS MaxPrice
236 FROM Sale
237 GROUP BY Category;
238
```

Result Grid




 Filter Rows:

Export: 

Wrap Cell Content: 

	Category	MaxPrice
▶	Appliances	484.79
	Accessories	483.05
	Furniture	492.18
	Clothing	491.46
	Electronics	422.75

Result 56 

Output



Action Output

▼

	#	Time	Action	Message
✓	76	15:49:49	SELECT * FROM Sale ORDER BY Price DESC LIMIT 0, 1000	200 row(s) returned
✓	77	15:50:16	SELECT Product, SUM(Quantity) AS TotalQuantity FROM Sale GROUP BY Product LIMIT 0...	10 row(s) returned
✓	78	15:50:49	SELECT AVG(Price) AS AveragePrice FROM Sale LIMIT 0, 1000	1 row(s) returned
✓	79	15:51:46	SELECT Category, MAX(Price) AS MaxPrice FROM Sale GROUP BY Category LIMIT 0, 1000	5 row(s) returned

```
238
239 -- 26) Retrieve Product and Quantity where Quantity is greater than the average Quantity of all products.
240 • SELECT Product, Quantity
241 FROM Sale
242 WHERE Quantity > (SELECT AVG(Quantity) FROM Sales);
243
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Product	Quantity
▶	Headphones	48
	Keyboard	65
	Mouse	68
	Camera	68
	Mouse	84
	Keyboard	88

Sale 57

Output

Action Output

#	Time	Action	Message
✓ 77	15:50:16	SELECT Product, SUM(Quantity) AS TotalQuantity FROM Sale GROUP BY Product LIMIT 0...	10 row(s) returned
✓ 78	15:50:49	SELECT AVG(Price) AS AveragePrice FROM Sale LIMIT 0, 1000	1 row(s) returned
✓ 79	15:51:46	SELECT Category, MAX(Price) AS MaxPrice FROM Sale GROUP BY Category LIMIT 0, 1000	5 row(s) returned
✓ 80	15:52:04	SELECT Product, Quantity FROM Sale WHERE Quantity > (SELECT AVG(Quantity) FROM ...	102 row(s) returned

```
252 -- 28) Select the number of records for each Category.
253 • SELECT Category, COUNT(*) AS RecordCount
254 FROM Sale
255 GROUP BY Category;
256
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Category	RecordCount
▶	Appliances	38
	Accessories	42
	Furniture	40
	Clothing	50
	Electronics	30

Result 58

Output

Action Output

	#	Time	Action	Message
✓	78	15:50:49	SELECT AVG(Price) AS AveragePrice FROM Sale LIMIT 0, 1000	1 row(s) returned
✓	79	15:51:46	SELECT Category, MAX(Price) AS MaxPrice FROM Sale GROUP BY Category LIMIT 0, 1000	5 row(s) returned
✓	80	15:52:04	SELECT Product, Quantity FROM Sale WHERE Quantity > (SELECT AVG(Quantity) FROM ...	102 row(s) returned
✓	81	15:52:40	SELECT Category, COUNT(*) AS RecordCount FROM Sale GROUP BY Category LIMIT 0, 1...	5 row(s) returned

256

257 -- 29) Find the Product with the highest Quantity and its corresponding Price.

258 • **SELECT** Product, Price

259 **FROM** Sale

260 **WHERE** Quantity = (**SELECT** MAX(Quantity) **FROM** Sale);

261

262 -- 30) Find the Product with the lowest Price and the total number of such products

**Result Grid** |  Filter Rows:  | Export:  | Wrap Cell Content: 

	Product	Price
▶	Monitor	484.79
	Monitor	484.79

Sale 59 x

Output



 Action Output ▼

	#	Time	Action	Message
✓	79	15:51:46	SELECT Category, MAX(Price) AS MaxPrice FROM Sale GROUP BY Category LIMIT 0, 1000	5 row(s) returned
✓	80	15:52:04	SELECT Product, Quantity FROM Sale WHERE Quantity > (SELECT AVG(Quantity) FROM ...	102 row(s) returned
✓	81	15:52:40	SELECT Category, COUNT(*) AS RecordCount FROM Sale GROUP BY Category LIMIT 0, 1...	5 row(s) returned
✓	82	15:53:04	SELECT Product, Price FROM Sale WHERE Quantity = (SELECT MAX(Quantity) FROM Sal...	2 row(s) returned

```

262 -- 30) Find the Product with the lowest Price and the total number of such products.
263 • SELECT Product, COUNT(*) AS Count
264 FROM Sale
265 WHERE Price = (SELECT MIN(Price) FROM Sale)
266 GROUP BY Product;
267


```

Result Grid |  Filter Rows:  | Export:  | Wrap Cell Content: 

	Product	Count
▶	Smartphone	2

Result 60 

Output 

 Action Output ▼

	#	Time	Action	Message
✓	80	15:52:04	SELECT Product, Quantity FROM Sale WHERE Quantity > (SELECT AVG(Quantity) FROM ...	102 row(s) returned
✓	81	15:52:40	SELECT Category, COUNT(*) AS RecordCount FROM Sale GROUP BY Category LIMIT 0, 1...	5 row(s) returned
✓	82	15:53:04	SELECT Product, Price FROM Sale WHERE Quantity = (SELECT MAX(Quantity) FROM Sal...	2 row(s) returned
✓	83	15:53:38	SELECT Product, COUNT(*) AS Count FROM Sale WHERE Price = (SELECT MIN(Price) F...	1 row(s) returned