

Limit to 50000 rows

Find

Done

1 • create database Intern;

2 • use intern;

3 • SELECT COUNT(*) FROM e_commerce;

4 -- total revenue

5 • SELECT SUM(Quantity*Unitprice) as Total_revenue from e_commerce;

6 -- total item sold

7 • select sum(Quantity) as TotalItesold from e_commerce;

8 -- Average Order Value

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

COUNT(*)
23795

Result 24

Result 25

Result 26

Result 27

Result 28

Result 29

Result 30

Result 31

Result 32

Result 33

Result 34

Result 35

Read Only

Find

Limit to 50000 rows

1 • create database Intern;

2 • use intern;

3 • SELECT COUNT(*) FROM e_commerce;

4 -- total revenue

5 • SELECT SUM(Quantity*Unitprice) as Total_revenue from e_commerce;

6 -- total item sold

7 • select sum(Quantity) as TotalItesold from e_commerce;

8 -- Average Order Value

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Total_revenue
▶	517569.88000002434

Result 24

Result 25

Result 26

Result 27

Result 28

Result 29

Result 30

Result 31

Result 32

Result 33

Result 34

Result 35



```

7 • select sum(Quantity) as TotalItemsold from e_commerce;
8 -- Average Order Value
9 • SELECT AVG(InvoiceTotal) AS AvgOrderValue
10 ○ FROM (
11     SELECT InvoiceNo, SUM(Quantity * UnitPrice) AS InvoiceTotal
12     FROM e_commerce
13     GROUP BY InvoiceNo
14 ) AS Subquery;

```

Result Grid Filter Rows: Export: Wrap Cell Content:

	AvgOrderValue
▶	408.17813880126204



```

13      GROUP BY InvoiceNo
14    ) AS Subquery;
15    -- top 10 customer
16    select CustomerID , sum(Quantity*UnitPrice) as TotalSpend from e_commerce
17    group by CustomerID order by TotalSpend limit 10;
18    -- top selling product
19    select Description , sum(Quantity) as TotalSold from e_commerce
20    group by Description order by TotalSold desc limit 10;

```

Result Grid   Filter Rows:  Export:  Wrap Cell Content:  Fetch rows:

	CustomerID	TotalSpend
▶	12476	12.45
	15823	15
	17816	16.2099999999999997
	17888	29.43
	18225	30
	18037	31.2999999999999997
	17026	33.15
	14576	35.4000000000000006
	14865	37.2
	12855	38.1

Result 24	Result 25	Result 26	Result 27	Result 28	Result 29	Result 30	Result 31	Result 32	Result 33	Result 34	Result 35	Result 36
-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------



```
19 • select Description , sum(Quantity) as TotalSold from e_commerce
20 group by Description order by TotalSold desc limit 10;
```

	Description	TotalSold
▶	WORLD WAR 2 GLIDERS ASSTD DESIGNS	4896
	WHITE HANGING HEART T-LIGHT HOLDER	3374
	HAND WARMER BABUSHKA DESIGN	2979
	GROW A FLYTRAP OR SUNFLOWER IN TIN	2592
	MINI PAINT SET VINTAGE	2520
	CREAM HEART CARD HOLDER	2235
	PACK OF 72 RETROSPOT CAKE CASES	2228
	ASSORTED COLOUR BIRD ORNAMENT	2219
	RED HARMONICA IN BOX	2161
	BLACK HEART CARD HOLDER	2140

BANK*Task3

Find

Limit to 50000 rows

16 • select CustomerID , sum(Quantity*UnitPrice) as TotalSpend from e_commerce

17 group by CustomerID order by TotalSpend limit 10;

18 -- top selling product

19 • select Description , sum(Quantity) as TotalSold from e_commerce

20 group by Description order by TotalSold desc limit 10;

21 • select Description , sum(Quantity*UnitPrice) as Revenue from e_commerce

22 group by Description order by Revenue desc limit 10;

23 -- sales by country

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Fetch rows:

Description	Revenue
▶ REGENCY CAKESTAND 3 TIER	16529.850000000006
WHITE HANGING HEART T-LIGHT HOLDER	8954.1
VINTAGE UNION JACK MEMOBOARD	6898.689999999997
WOOD BLACK BOARD ANT WHITE FINISH	6502.530000000002
BLACK RECORD COVER FRAME	6022.35
CHILLI LIGHTS	5848.790000000002
CREAM HEART CARD HOLDER	5461.530000000001
BLACK HEART CARD HOLDER	5106.440000000005
ROUND SNACK BOXES SET OF 4 FRUITS	4713.949999999997
PAPER CHAIN KIT 50'S CHRISTMAS	4464.949999999998

Result 24

Result 25

Result 26

Result 27

Result 28

Result 29

Result 30 ×

Result 31

Result 32

Result 33

Result 34

Result 35

Re

Limit to 50000 rows

Find

22 group by Description order by Revenue desc limit 10;

23 -- sales by country

24 • select Country , SUM(Quantity*Unitprice) as revenue from e_commerce

25 group by Country order by revenue desc;

26

27 -- Daily Sales Trend

28 • SELECT DATE(InvoiceDate) AS Date, SUM(Quantity * UnitPrice) AS DailySales

29 FROM e_commerce GROUP BY Date ORDER BY Date;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Country	revenue
▶	United Kingdom	459993.09000001964
	Germany	12747.760000000002
	France	7913.439999999991
	Japan	7705.0700000000015
	EIRE	7350.629999999998
	Sweden	3834.3
	Norway	3787.1199999999994
	Portugal	2439.9699999999984
	Spain	1843.7299999999998
	Lithuania	1661.06
	Country	1500.87

Result 24

Result 25

Result 26

Result 27

Result 28

Result 29

Result 30

Result 31

Result 32

Result 33

Result 34

Result 35

Read

In
fo

Res
Gr

For
Edi

Fie
Typ

Find

Limit to 50000 rows

77 •

UPDATE e_commerce

78

SET InvoiceDate_new = STR_TO_DATE(InvoiceDate, '%m/%d/%Y %H:%i');

79 •

SELECT InvoiceDate, InvoiceDate_new FROM e_commerce LIMIT 10;

80 •

ALTER TABLE e_commerce DROP COLUMN InvoiceDate;

81 •

ALTER TABLE e_commerce CHANGE COLUMN InvoiceDate_new InvoiceDate DATETIME;

82

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Fetch rows:

	InvoiceDate	InvoiceDate_new
▶	12/1/2010 8:26	2010-12-01 08:26:00
	12/1/2010 8:26	2010-12-01 08:26:00
	12/1/2010 8:26	2010-12-01 08:26:00
	12/1/2010 8:26	2010-12-01 08:26:00
	12/1/2010 8:26	2010-12-01 08:26:00
	12/1/2010 8:26	2010-12-01 08:26:00
	12/1/2010 8:26	2010-12-01 08:26:00
	12/1/2010 8:28	2010-12-01 08:28:00
	12/1/2010 8:28	2010-12-01 08:28:00

e_commerce 44

Output

Navigation icons: back, forward, search, and a small icon.

```
33 • SELECT DATE_FORMAT(InvoiceDate, '%Y-%m') AS Month,
```

```
35 FROM e_commerce GROUP BY Month ORDER BY Month;
```

Result Grid

Filter Rows:

Export:



Wrap Cell Content: 

	Month	MonthlySales
▶	2010-12	517569.88000002434

Result 46

Result 48

Result 50


Result 52

Result 53


```

41
42
43 -- Revenue per Invoice
44 • SELECT InvoiceNo, SUM(Quantity * UnitPrice) AS InvoiceTotal
45 FROM e_commerce GROUP BY InvoiceNo ORDER BY InvoiceTotal DESC;
46 -- Items per Invoice
47 • SELECT InvoiceNo, SUM(Quantity) AS ItemsPerInvoice
48 FROM e commerce GROUP BY InvoiceNo;

```

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

InvoiceNo	InvoiceTotal
537659	15885.489999999998
537657	9639.119999999999
539101	6929.999999999999
538191	5375.28
536783	4076.48
537899	3794.3999999999996
538353	3762.66
538397	3590.5900000000006
538991	3240.3999999999999
536387	3193.92
537703	2746.7400000000000

Result 45 Result 46 Result 47 **Result 48** × Result 49 Result 50 Result 51 Result 52 Result 53

Find

Limit to 50000 rows

44 • SELECT InvoiceNo, SUM(Quantity * UnitPrice) AS InvoiceTotal

45 FROM e_commerce GROUP BY InvoiceNo ORDER BY InvoiceTotal DESC;

46 -- Items per Invoice

47 • SELECT InvoiceNo, SUM(Quantity) AS ItemsPerInvoice

48 FROM e_commerce GROUP BY InvoiceNo;

49 -- Repeat Customers

50 • SELECT CustomerID, COUNT(DISTINCT InvoiceNo) AS NumberOfOrders

51 FROM e commerce

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	InvoiceNo	ItemsPerInvoice
▶	536365	40
	536366	12
	536367	83
	536368	15
	536369	3
	536370	449
	536371	80
	536372	12
	536373	88
	536374	32
	536375	88

Result 45

Result 46

Result 47

Result 48

Result 49 ×

Result 50

Result 51

Result 52

Result 53

Output

C


```
47 • SELECT InvoiceNo, SUM(Quantity) AS ItemsPerInvoice
48 FROM e_commerce GROUP BY InvoiceNo;
49 -- Repeat Customers
50 • SELECT CustomerID, COUNT(DISTINCT InvoiceNo) AS NumberOfOrders
51 FROM e_commerce
52 GROUP BY CustomerID
53 HAVING NumberOfOrders > 1
54 ORDER BY NumberOfOrders DESC;
```


Filter Rows:

Export:



Wrap Cell Content:

 \overline{IA}

	CustomerID	NumberOfOrders
►	17850	34
	12748	30
	15061	12
	14911	10
	13777	9
	13089	8
	15311	7
	13767	6
	14606	6
	13408	5
	13418	5

Result 46

Result 47

Result 48

Result 49

Result 50 

Result 51

Result 52

Result 53

Output

BANK*

Task3*



```
61
62
63  -- High-Spending Customers
64  • SELECT
65      CASE
66          WHEN OrderCount = 1 THEN 'One-time'
67          ELSE 'Repeat'
68      END AS CustomerType,
69      COUNT(*) AS NumCustomers
70  FROM /
```

Result Grid



Filter Rows:

Export:



Wrap Cell Content:

	CustomerType	NumCustomers
►	One-time	601
	Repeat	235