

class 5 class 6 class 7 class 8 mobiles class 9* movies movies class 10 employees partice 1 class 11 my bank* x

Limit to 50000 rows

```
4
5 -- retrieve all table data
6 • select * from customers;
7 • select * from atms;
8 • select * from accounts;
9 • select * from transactions;
10 • select * from loans;
11 • select * from credit_cards;
12 • select * from branches;
13 • select * from customers,atms,accounts,transactions,loans,credit_cards,branches;
14
```

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

	CustomerID	Name	Age	Gender	Address	PhoneNumber	Email	ATMID	Location	InstalledDate	Status	
▶	1	James Williams	21	Female	651 Lake St, Philadelphia, PA	12839232977	jameswilliams@yahoo.com	2	970 Elm St, Anytown, USA	2022-04-25	Operational	1
	1	James Williams	21	Female	651 Lake St, Philadelphia, PA	12839232977	jameswilliams@yahoo.com	3	66 Main St, Anytown, USA	2022-11-11	Operational	1
	1	James Williams	21	Female	651 Lake St, Philadelphia, PA	12839232977	jameswilliams@yahoo.com	4	153 Elm St, Anytown, USA	2023-03-20	Operational	1
	1	James Williams	21	Female	651 Lake St, Philadelphia, PA	12839232977	jameswilliams@yahoo.com	5	487 Park Ave, Anytown, USA	2023-08-15	Operational	1
	1	James Williams	21	Female	651 Lake St, Philadelphia, PA	12839232977	jameswilliams@yahoo.com	6	63 Elm St, Anytown, USA	2022-07-09	Out of Service	1


Result Grid Form Editor


```
21
22 -- calculate total loan amount
23 • select sum(amount) as totalloanamounts from loans;
24
25 -- calculate total credit limit across all credit cards
26 • select sum(creditlimit) as totalcreditlimit from credit_cards;
27
28 -- find all active account
29 • select * from accounts where status='active';
30
31 -- final all account made on 15 jan 2023
```

Result Grid		Filter Rows:	Exports:	Wrap Cell Contents: TA		
	AccountNumber	CustomerID	AccountType	Balance	OpenDate	Status
▶	1	92	Investment	38787.46	24-07-2020	Active
	4	48	Savings	26641.48	12-10-2020	Active
	5	55	Checking	9562.33	24-01-2020	Active
	7	36	Checking	7848.57	24-09-2022	Active
	8	45	Checking	23380.26	24-03-2021	Active
	17	33	Savings	17492.59	03-11-2020	Active

#	Time	Action	Message	Duration / Fetch
4	17:45:11	select * from customers,atms,accounts,transactions,loans,credit_cards,branches LIMIT 0, ...	50000 row(s) returned	0.062 sec / 0.500 sec
5	18:05:11	select count(*) as totalcustomers from customers LIMIT 0, 50000	1 row(s) returned	0.000 sec / 0.000 sec
6	18:08:39	select * from accounts where status='active' LIMIT 0, 50000	50 row(s) returned	0.000 sec / 0.000 sec

class 5class 6class 7class 8mobilesclass 9"moviesmoviesclass 10employeespartice 1class 11 my bank" x


Limit to 50000 rows

```
21
22 -- calculate total loan amount
23 • select sum(amount) as totalloanamounts from loans;
24
25 -- calculate total credit limit across all credit cards
26 • select sum(creditlimit) as totalcreditlimit from credit_cards;
27
28 -- find all active account
29 • select * from accounts where status='active';
30
31 -- final all account made on 15 jan 2023
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	AccountNumber	CustomerID	AccountType	Balance	OpenDate	Status
▶	1	92	Investment	38787.46	24-07-2020	Active
	4	48	Savings	26641.48	12-10-2020	Active
	5	55	Checking	9562.33	24-01-2020	Active
	7	36	Checking	7848.57	24-09-2022	Active
	8	45	Checking	23380.26	24-03-2021	Active
	17	33	Savings	17492.59	03-11-2020	Active

accounts 3 x

Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 4	17:45:11	select * from customers,atms,accounts,transactions,loans,credit_cards,branches LIMIT 0, ...	50000 row(s) returned	0.062 sec / 0.500 sec
✓ 5	18:05:11	select count(*) as totalcustomers from customers LIMIT 0, 50000	1 row(s) returned	0.000 sec / 0.000 sec
✓ 6	18:08:39	select * from accounts where status='active' LIMIT 0, 50000	50 row(s) returned	0.000 sec / 0.000 sec

class 5 class 6 class 7 class 8 mobiles class 9* movies movies class 10 employees partice 1 class 11 my bank* x

Limit to 50000 rows

```
59 -- delete inactive accounts
60 • set sc Execute the statement under the keyboard cursor
61 • delete from accounts
62 where status='inactive';
63
64 -- find customers with multiple account
65 • select c.customerid,c.name,count(a.accountnumber) as numaccounts
66 from customers c
67 join accounts a on c.customerid=a.customerid
68 group by c.customerid,c.name
69 having count(a.accountnumber)>1;
```

Result Grid Filter Rows: Export: Wrap Cell Content: [F6](#)

	customerid	name	numaccounts
▶	7	Emily Jones	4
	29	John Johnson	2
	31	William Jones	2
	34	William Martinez	2
	43	James Smith	2
	46	William Johnson	2

Result 6 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓	7 18:13:50	select * from credit_cards where balance>creditlimit LIMIT 0, 50000	0 row(s) returned	0.000 sec / 0.000 sec
✓	8 18:32:10	select t.transactionid,t.transactiondate,t.amount,t.type,t.description,a.accountnumber,a.ac...	47 row(s) returned	0.000 sec / 0.000 sec
✓	9 18:34:08	select c.customerid,c.name,count(a.accountnumber) as numaccounts from customers c jo...	10 row(s) returned	0.016 sec / 0.000 sec

Activate Windows
Go to Settings to activate Windows.

The screenshot shows the Oracle SQL Developer interface. The top pane contains a SQL script with three queries. The bottom pane shows the 'Result Grid' for the third query, displaying a table with first and last names. The 'Output' pane at the bottom shows the execution log with timestamps and messages for each query.

SQL Script:

```

71  -- print first 3 characters of name from customers table
72  • select substr(name,1,3) as firstthreecharactersofname
73  from customers;
74
75  -- print the name from coustomer table into two columns fristname and lastname
76  • select
77  substr_index(name, ' ',1)as firstname,
78  substr_index(name, ' ',-1)as lastname
79  from customers;
80
81  -- sql query to show only odd rows from customers table

```

Result Grid:

firstname	lastname
James	Williams
William	Brown
John	Miller
Isabella	Jones
Jane	Davis
James	Johnson

Output:

#	Time	Action	Message	Duration / Fetch
8	18:32:10	select t.transactionid,t.transactiondate,t.amount,t.type,t.description,a.accountnumber,a.ac...	47 row(s) returned	0.000 sec / 0.000 sec
9	18:34:08	select c.customerid,c.name,count(a.accountnumber) as numaccounts from customers c jo...	10 row(s) returned	0.016 sec / 0.000 sec
10	18:36:04	select substr_index(name,' ',1)as firstname, substr_index(name,' ',-1)as lastname from...	100 row(s) returned	0.000 sec / 0.000 sec

class 5 class 6 class 7 class 8 mobiles class 9* movies movies class 10 employees partice 1 class 11 my bank

Limit to 50000 rows

```
79 from customers;
80 -- sql query to show only odd rows from customers table
81 select * from customers
82 where mod(customerid,2)<>0;
83
84 -- sql query to determine the 5th highest loan amount without using limit keyword
85 select distinct amount
86 from loans l1
87 where 5=(
88 select count(distinct amount)
89
```

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

	CustomerID	Name	Age	Gender	Address	PhoneNumber	Email
▶	1	James Williams	21	Female	651 Lake St, Philadelphia, PA	12839232977	jameswilliams@yahoo.com
	3	John Miller	69	Female	258 Elm St, San Antonio, PA	19582400760	johnmiller@aol.com
	5	Jane Davis	59	Male	403 Main St, Houston, TX	14674802399	janedavis@aol.com
	7	Emily Jones	23	Female	75 Park Ave, San Diego, NY	15713857589	emilyjones@yahoo.com
	9	Emily Rodriguez	51	Male	862 Washington St, Chicago, AZ	15104811059	emilyrodriguez@aol.com
	11	William Johnson	22	Male	129 Maple Ave, Los Angeles, CA	15810002554	williamjohnson@aol.com

customers 8 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓	9 18:34:08	select c.customerid,c.name,count(a.accountnumber) as numaccounts from customers c jo...	10 row(s) returned	0.016 sec / 0.000 sec
✓	10 18:36:04	select substring_index(name,' ',1)as firstname, substring_index(name,' ',-1)as lastname from...	100 row(s) returned	0.000 sec / 0.000 sec
✓	11 18:38:00	select * from customers where mod(customerid,2)<>0 LIMIT 0, 50000	50 row(s) returned	0.000 sec / 0.000 sec

Activate Windows
Go to Settings to activate Windows.

class 5class 6class 7class 8mobilesclass 9*moviesmoviesclass 10employeespartice 1class 11 my bank

Limit to 50000 rows

117 -- sql query to calculate how many days are remaing for customers to pay off the loans
118 • select
119 customerid,
120 datediff(enddate,curdate()) as daysremaining
121 from loans
122 where enddate>curdate();
123
124 -- query to find the latest transaction date for each account
125 • select accountnumber,max(transactiondate) as latesttransactiondate
126 from transactions
127 group by accountnumber;
128
129 -- find the avergae age of customers
130 • select avg(age) as averageage



Result GridFilter Rows:Export:Wrap Cell Content: 1A

	accountnumber	latesttransactiondate
▶	98	2023-07-26
	60	2023-02-17
	3	2022-07-24
	32	2023-04-13
	41	2023-06-03
	89	2022-06-09
	5	2023-08-29
	63	2023-04-26
	91	2023-09-01




Result 9 x

Activate Windows
Go to Settings to activate Windows.
Read Only

class 5class 6class 7class 8mobilesclass 9*moviesmoviesclass 10employeespartice 1class 11 my bank* x

Limit to 50000 rows




```
133 -- find accounts with less than miniumum amount for accounts opened before 1st jan 2022
134 • select accountnumber,balance
135 from accounts
136 where balance<25000
137 and opendate<='2022-01-01';
138
139 -- find loan that are currently active
140 • select*
141 from loans
142 where enddate>=curdate()
143 and status='active';
144
145 -- find the total amount of transactions for each account for a specific month
146 • select accountnumber,sum(amount) as totalamount
```

Result Grid  Filter Rows: Export:  Wrap Cell Content: 

	accountnumber	latesttransactiondate
▶	98	2023-07-26
	60	2023-02-17
	3	2022-07-24
	32	2023-04-13
	41	2023-06-03
	89	2022-06-09
	5	2023-08-29
	63	2023-04-26
	91	2023-09-01

Result 9 x

Activate Windows
Go to Settings to activate Windows.

Result Grid
Form Editor
Read Only

```
156
157 -- find the number of inactive atms per location
158 • select location,count(*) as numberofactiveatms
159 from atms where status ='out of service'
160 group by location;
161
162 -- categorize customers into three agr groups
163 • select name, age ,
164 case
165 when age<30 then 'below 30'
166 when age between 30 and 60 then '30 to 60'
167 else 'above 60'
168 end as age_group
169 from customers;
```

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

	name	age	age_group
▶	James Williams	21	below 30
	William Brown	41	30 to 60
	John Miller	69	above 60
	Isabella Jones	64	above 60
	Jane Davis	59	30 to 60
	James Johnson	65	above 60
	Emily Jones	23	below 30
	Michael Brown	47	30 to 60
	Emilv Rodriguez	51	30 to 60

Result 10 x

Activate Windows
Go to Settings to activate Windows

Read Only