

MySQL Internal Assessment

Date: 29.11.2023

Name: Karella Rupa Naga Prasanna Raji

1. Write a mysql statement to find the concatenated first_name, last_name where the age of the employee is greater than 30. Suppose the employee table is -

first_name	last_name	age	dept
Mesa	Loop	30	Acct
Smith	Oak	27	Dev1
John	Jorz	37	QA
Hary	Gaga	32	QA

Query: select concat(first_name, ' ', last_name) as Name from employees where age>30

The screenshot shows a MySQL IDE interface with a SQL editor and a results pane. The SQL editor contains the following queries:

```
1 • use hexaware;
2
3 • create table employees(first_name varchar(10), last_name varchar(10), age int, dept varchar(5))
4 • insert into employees values('Mesa', 'Loop', 30, 'Acct');
5 • insert into employees values('Smith', 'Oak', 27, 'Dev1');
6 • insert into employees values('John', 'Jorz', 37, 'QA');
7 • insert into employees values('Hary', 'Gaga', 32, 'QA');
8 • select * from employees;
9
10 • select concat(first_name, ' ', last_name) as Name from employees where age>30
```

The results pane shows the output of the queries. The first query (select * from employees) returned 4 rows. The second query (select concat(first_name, ' ', last_name) as Name from employees where age>30) returned 2 rows, which are John Jorz and Hary Gaga.

Name
John Jorz
Hary Gaga

Result 2 x Read Only

Output

#	Time	Action	Message
27	16:32:18	select * from employees LIMIT 0, 1000	4 row(s) returned
28	16:34:31	select concat(first_name, ' ', last_name) as Name from employees where age>30 LIMIT 0, 10...	2 row(s) returned

2. Write a mysql statement to get the user, current date and mysql version.

Query: select current_user() as CurrentUser, current_date() as CurrentDate, version() as SQLVersion;

The screenshot shows a MySQL IDE interface. The SQL editor contains the following code:

```
13 -- 2. get user, current date and mysql version
14 • select current_user() as CurrentUser, current_date() as CurrentDate, version() as SQLVersion
```

Below the editor, the 'Result Grid' shows the output of the query:

CurrentUser	CurrentDate	SQLVersion
root@localhost	2023-11-29	8.0.34

The 'Output' pane shows the execution log:

#	Time	Action	Message
29	16:42:15	select current_user(), current_date(), version() LIMIT 0, 1000	1 row(s) returned
30	16:43:14	select current_user() as CurrentUser, current_date() as CurrentDate, version() as SQLVersion ...	1 row(s) returned

3. Write a mysql statement to get an item id, item, price of the most expensive item

Query: select * from items where price=(select max(price) from items);

The screenshot shows a MySQL IDE interface. The SQL editor contains the following code:

```
16 -- 3. get an item id, item, price of the most expensive item
17 • create table items(item_id int, item varchar(10), price int);
18 • insert into items values(1, 'Laptop', 65000);
19 • insert into items values(2, 'Camera', 55000);
20 • insert into items values(3, 'Earphones', 3000);
21 • insert into items values(4, 'SmartWatch', 6000);
22 • insert into items values(5, 'Mobile', 92000);
23 • insert into items values(6, 'SmartTV', 125000);
24 • select * from items;
25 • select * from items where price=(select max(price) from items);
26
```

Below the editor, the 'Result Grid' shows the output of the query:

item_id	item	price
6	SmartTV	125000

The 'Output' pane shows the execution log:

#	Time	Action	Message
43	16:56:15	select * from items order by price desc limit 1	1 row(s) returned
44	16:57:56	select * from items where price=(select max(price) from items) LIMIT 0, 1000	1 row(s) returned

4. Write a mysql statement to create a new user and set a password and privileges for an existing database.

Query: create user 'rupa'@'localhost' identified by 'Rupa@123';

grant all privileges on hexaware to rupa@localhost;

```
26 -- 4. create a new user and set a password and privileges for an existing database
27 • create user 'rupa'@'localhost' identified by 'Rupa@123';
28 • grant all privileges on hexaware to rupa@localhost;
29 • show grants for rupa@localhost;
```

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

	Grants for rupa@localhost
▶	GRANT USAGE ON *.* TO 'rupa'@'localhost'
	GRANT ALL PRIVILEGES ON 'hexaware'. 'hexa...

Result 9 x Read Or

Output			
Action Output			
#	Time	Action	Message
✓ 45	17:12:18	create user 'rupa'@'localhost' identified by 'Rupa@123'	0 row(s) affected
✓ 46	17:12:18	grant all privileges on hexaware to rupa@localhost	0 row(s) affected
✓ 47	17:13:25	show grants for rupa@localhost	2 row(s) returned

5. Write a mysql statement to select data of only CS and IT departments. Suppose the table is -

id	name	department	age
1	Maria Gloria	CS	22
2	John Smith	IT	23
3	Gal Rao	CS	22
4	Jakey Smith	EC	24
5	Rama Saho	IT	22
6	Maria Gaga	EC	23

Query: select * from students where department='CS' or department='IT';

```
31 -- 5. select data of only CS and IT departments
32 • create table students(id int, name varchar(15), department varchar(5),age int);
33 • insert into students values(1,'Maria Gloria','CS',22);
34 • insert into students values(2, 'John Smith', 'IT', 23);
35 • insert into students values(3, 'Gal Rao', 'CS', 22);
36 • insert into students values(4, 'Jakey Smith', 'EC', 24);
37 • insert into students values(5, 'Rama Saho', 'IT', 22);
38 • insert into students values(6, 'Maria Gaga', 'EC', 23);
39 • select * from students where department='CS' or department='IT';
40
```

id	name	department	age
1	Maria Gloria	CS	22
2	John Smith	IT	23
3	Gal Rao	CS	22
5	Rama Saho	IT	22

students 10 × Read Only

Output

#	Time	Action	Message
✓ 54	17:36:42	insert into students values(5, 'Rama Saho', 'IT', 22)	1 row(s) affected
✓ 55	17:36:42	insert into students values(6, 'Maria Gaga', 'EC', 23)	1 row(s) affected
✓ 56	17:36:42	select * from students where department='CS' or department='IT' LIMIT 0, 1000	4 row(s) returned

6. Write a mysql statement to select data of all departments in descending order by age.

Query: select * from students order by age desc;

```
41 -- 6.select data of all departments in descending order by age.
42 • select * from students order by age desc;
```

id	name	department	age
4	Jakey Smith	EC	24
2	John Smith	IT	23
6	Maria Gaga	EC	23
1	Maria Gloria	CS	22
3	Gal Rao	CS	22

students 11 × Read Only

Output

#	Time	Action	Message
✓ 55	17:36:42	insert into students values(6, 'Maria Gaga', 'EC', 23)	1 row(s) affected
✓ 56	17:36:42	select * from students where department='CS' or department='IT' LIMIT 0, 1000	4 row(s) returned
✓ 57	17:43:30	select * from students order by age desc LIMIT 0, 1000	6 row(s) returned

7. Write a mysql statement to determine the age of each of the students. Suppose the table is -

id	name	department	birth
1	Maria Gloria	CS	1994-03-12
2	John Smith	IT	1993-02-07
3	Gal Rao	CS	1992-09-11
4	Jakey Smith	EC	1990-08-31
5	Rama Saho	IT	1994-12-09
6	Maria Gaga	EC	1993-10-09

Query: select *, year(current_date())-year(birth) as age from students1;

The screenshot shows a MySQL IDE window with a query editor and a results grid. The query editor contains the following SQL code:

```
-- 7. to determine the age of each of the students
create table students1(id int, name varchar(15), department varchar(5), birth varchar(10));
insert into students1 values(1, 'Maria Gloria', 'CS', '1994-03-12');
insert into students1 values(2, 'John Smith', 'IT', '1993-02-07');
insert into students1 values(3, 'Gal Rao', 'CS', '1992-09-11');
insert into students1 values(4, 'Jakey Smith', 'EC', '1990-08-31');
insert into students1 values(5, 'Rama Saho', 'IT', '1994-12-09');
insert into students1 values(6, 'Maria Gaga', 'EC', '1993-10-09');
select *, year(current_date())-year(birth) as age from students1;
```

The results grid displays the following data:

id	name	department	birth	age
1	Maria Gloria	CS	1994-03-12	29
2	John Smith	IT	1993-02-07	30
3	Gal Rao	CS	1992-09-11	31
4	Jakey Smith	EC	1990-08-31	33
5	Rama Saho	IT	1994-12-09	29
6	Maria Gaga	EC	1993-10-09	30

The output pane shows the execution of the query, indicating that 6 row(s) were returned.

8. Write a mysql statement to retrieve names beginning with 'm'.

Query: select * from students1 where name like 'm%';

The screenshot shows a MySQL IDE window with a query editor and a results grid. The query editor contains the following SQL code:

```
-- 8. statement to retrieve name beginning with 'm'.
select * from students1 where name like 'm%';
```

The results grid displays the following data:

id	name	department	birth
1	Maria Gloria	CS	1994-03-12
6	Maria Gaga	EC	1993-10-09

The output pane shows the execution of the query, indicating that 2 row(s) were returned.

9. Write a mysql statement to find the name, birth, department name, department block from the given tables.

id	name	dept_id	birth
1	Maria Gloria	2	1994-03-12
2	John Smith	1	1993-02-07
3	Gal Rao	4	1992-09-11
4	Jakey Smith	2	1990-08-31
5	Rama Saho	1	1994-12-09
6	Maria Gaga	4	1993-10-09

dept_id	dept_name	dept_block
1	Computer Science	B-Block
2	Information Technology	C-Block
3	Mechanical	A-Block
4	Electronic Communication	D-Block

Query- select students2.name, students2.birth, department.deptName, department.deptBlock from students2 left join department on students2.deptid=department.deptid;

```
71 -- 9. to find the name, birth, department name, department block from the given tables.
72 • select students2.name, students2.birth, department.deptName, department.deptBlock
73 from students2 left join department on students2.deptid=department.deptid;
```

name	birth	deptName	deptBlock
Maria Gloria	1994-03-12	Information Technology	C-Block
John Smith	1993-02-07	Computer Science	B-Block
Gal Rao	1992-09-11	Electronic Communication	D-Block
Jakey Smith	1990-08-31	Information Technology	C-Block
Rama Saho	1994-12-09	Computer Science	B-Block
Maria Gaga	1993-10-09	Electronic Communication	D-Block

Result 17 ×

Read Only Cc

Output

Action Output

#	Time	Action	Message
89	21:07:53	insert into department values(4,'Electronic Communication','D-Block')	1 row(s) affected
90	21:07:53	select students2.name, students2.birth, department.deptName, department.deptBlock from s...	6 row(s) returned

10. Write a mysql statement to get the names of students containing exactly four characters. Suppose the student table is -

id	name	dept_id	birth
1	Maria	2	1994-03-12
2	John	1	1993-02-07
3	Gal	4	1992-09-11
4	Jakey	2	1990-08-31
5	Rama	1	1994-12-09
6	Maria	4	1993-10-09

Query: select name from student where char_length(name)=4;

The screenshot shows a MySQL IDE window titled 'InternalAssessment_Rupa'. The SQL editor contains the following code:

```
-- 10. to get the names of students containing exactly four characters
• create table student(id int, name varchar(15), deptid int,birth varchar(10));
• insert into student values(1,'Maria','2','1994-03-12');
• insert into student values(2, 'John', '1', '1993-02-07');
• insert into student values(3, 'Gal', '4', '1992-09-11');
• insert into student values(4, 'Jakey', '2', '1990-08-31');
• insert into student values(5, 'Rama', '1', '1994-12-09');
• insert into student values(6, 'Maria', '4', '1993-10-09');
• select name from student where char_length(name)=4;
```

The 'Result Grid' shows the output of the query:

name
John
Rama

The 'Output' pane shows the execution log:

#	Time	Action	Message
97	21:21:31	insert into student values(6, 'Maria', '4', '1993-10-09')	1 row(s) affected
98	21:21:31	select name from student where char_length(name)=4 LIMIT 0, 1000	2 row(s) returned

11. Write a mysql statement to delete duplicate row from the table

Query- delete b1 from book b1 join(select bookName, price from book group by bookName, price having count(*) > 1) b2 on b1.bookName = b2.bookName and b1.price = b2.price;

```
94 • delete b1 from book b1
95   join(
96     select bookName, price from book group by bookName, price having count(*) > 1
97   ) b2
98   on b1.bookName = b2.bookName and b1.price = b2.price;
99
```

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

	bookName	price
▶	Life Switch	82
	Who moved my cheese	120

book 28 x | Read Or

Output

Action Output

#	Time	Action	Message
✓ 162	22:44:36	select * from book LIMIT 0, 1000	2 row(s) returned
✓ 163	22:45:23	select * from book LIMIT 0, 1000	2 row(s) returned

12. Write a mysql statement to display the alternate rows from MySQL table.

Query- select * from student where id%2=0;

```
101  -- 12. display alternate records
102 • select * from student where id%2=0;
103
```

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

	id	name	deptid	birth
▶	2	John	1	1993-02-07
	4	Jakey	2	1990-08-31
	6	Maria	4	1993-10-09

student 38 x | Reac

Output

Action Output

#	Time	Action	Message
✓ 182	23:32:13	insert into student values(6, 'Maria', '4', '1993-10-09')	1 row(s) affected
✓ 183	23:32:32	select * from student where id%2=0 LIMIT 0, 1000	3 row(s) returned

13. Write a mysql statement to delete alternate records

Query- delete from table where id%2=0;

```
104 -- 13. delete alternate rows
105 • delete from student where id%2=0;
106
107 -- 14. update multiple rows in one query
```

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

	id	name	deptid	birth
▶	1	Maria	2	1994-03-12
	3	Gal	4	1992-09-11
	5	Rama	1	1994-12-09

student 39 x

Output

Action Output

#	Time	Action	Message
✓ 184	23:33:18	delete from student where id%2=0	3 row(s) affected
✓ 185	23:33:28	select * from student LIMIT 0, 1000	3 row(s) returned

14. Write a mysql statement to update multiple rows in a single query

Query- update emp set sal=18000 where dept='Sales';

```
107 -- 14. update multiple rows in one query
108 update emp set sal=18000 where dept='Sales';
109 • select * from emp where dept='Sales';
110
```

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

	eid	ename	sal	loc	dept
▶	579	Ali	18000	Hyd	Sales
	520	Seetha	18000	Mum	Sales
	782	Ali	18000	Hyd	Sales
	806	Siva	18000	Bnglr	Sales
	165	Janvi	18000	Mum	Sales

emp 32 x

Read Only

Output

Action Output

#	Time	Action	Message
✓ 168	23:11:35	update emp set sal=18000 where dept='Sales'	6 row(s) affected Rows matched: 6 Changed: 6 Warnings: 0
✓ 169	23:11:35	select * from emp where dept='Sales' LIMIT 0, 1000	6 row(s) returned

15. Write a mysql statement

a. to get nth highest paid

Query- [select * from employee order by annSal desc limit 1 Offset n-1;]

select * from employee order by annSal desc limit 1 Offset 5; -- say n=6

```
111 -- 15. nth highest paid salary
112 -- select * from employee order by annSal desc limit 1 Offset n-1;
113 • select * from employee order by annSal desc limit 1 Offset 5; -- say n=6
114
```

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

	eID	eName	loc	dept	annSal	yoj	mgrID
▶	1	Anil	Hyd	IT	700000	2017	6

employee35 × Read Only

Output

Action Output ▼

#	Time	Action	Message
✓ 171	23:18:09	select * from employee order by annSal desc LIMIT 0, 1000	20 row(s) returned
✓ 172	23:18:22	select * from employee order by annSal desc limit 1 Offset 5	1 row(s) returned

b. to get nth least paid

Query- [select * from employee order by annSal limit 1 Offset n-1;]

select * from employee order by annSal limit 1 Offset 6; -- say n=7

```
115 -- 15. b) to get nth least paid
116 -- select * from employee order by annSal limit 1 Offset n-1;
117 • select * from employee order by annSal limit 1 Offset 4; -- say n=5
118
119
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Fetch rows:

eID	eName	loc	dept	annSal	yoj	mgrID
14	Ajay	Hyd	Mkt	400000	2021	3

employee 37

Output

Action Output

#	Time	Action	Message
173	23:30:28	select * from employee order by annSal limit 1 Offset 6	1 row(s) returned
174	23:30:43	select * from employee order by annSal limit 1 Offset 4	1 row(s) returned