SQL Assignment Solutions

Worker Database created

```
1 • CREATE DATABASE worker;
2
3 • SHOW DATABASES;
```

Result Grid | Filter Rows:

#	Database
1	courses
2	information_schema
3	mysql
4	performance_schema
5	practice
6	sys
7	worker

Creating Employee Table

```
1  • CREATE TABLE Employee (
2      emp_no int,
3      first_name varchar(25),
4      last_name varchar(25),
5      email varchar(40),
6      department varchar(25),
7      salary int,
8      join_date date,
9      primary key (emp_no)
10 );
11
12 • describe Employee;
13
```

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

#	Field	Type	Null	Key	Default	Extra
1	emp_no	int	NO	PRI	NULL	
2	first_name	varchar(25)	YES		NULL	
3	last_name	varchar(25)	YES		NULL	
4	email	varchar(40)	YES		NULL	
5	department	varchar(25)	YES		NULL	
6	salary	int	YES		NULL	
7	join_date	date	YES		NULL	

Populating 30 records

```
14 •  INSERT INTO Employee (emp_no, first_name, last_name, email, department, salary, join_date)
15    VALUES (1, 'Selvi', 'Jayaraman', 'selvi@gmail.com', 'Technical', '30000', '2019-07-09');
16
17
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
1	1	Selvi	Jayaraman	selvi@gmail.com	Technical	30000	2019-07-09
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
46 •  INSERT INTO Employee (emp_no, first_name, last_name, email, department, salary, join_date)
47    VALUES (21, 'Andrew', 'Faulkner', 'andrew@guvi.com', 'Development', '55000', '2016-08-05'),
48    (22, 'Karen', 'Mathew', 'karen@guvi.com', 'Salesman', '40000', '2017-11-12'),
49    (23, 'Wendy', 'Shawn', 'wendy@network.com', 'Testing', '35000', '2019-08-10'),
50    (24, 'Bella', 'Swan', 'bella@solutions.in', 'Development', '55000', '2013-08-09'),
51    (25, 'Kevin', 'Hill', 'hill@yahoo.com', 'IT support', '20000', '2019-07-09'),
52    (26, 'Madii', 'Himbury', 'madii@gmail.com', 'Technical', '30500', '2005-02-16'),
53    (27, 'Athena', 'Wilson', 'athena@yahoo.com', 'Analyst', '32000', '2015-04-10'),
54    (28, 'Jennifer', 'Huette', 'jeni@guvi.com', 'Manager', '80000', '2021-04-15'),
55    (29, 'John', 'Smith', 'josmi@yahoo.com', 'Engineer', '60000', '2009-04-12'),
56    (30, 'Alan', 'Mathew', 'mathew@gmail.com', 'IT support', '20000', '2017-06-12');
57
58 •  SELECT * FROM Employee;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
1	1	Selvi	Jayaraman	selvi@gmail.com	Technical	30000	2019-07-09
2	2	Ram	Sundar	ram@gmail.com	Technical	35000	2018-08-05
3	3	Alex	George	alex@yahoo.com	Finance	20000	2016-11-12
4	4	Alford	Francis	alford@gmail.com	Testing	25000	2010-07-09
5	5	Ravi	Kumar	kumar@solutions.in	Development	40000	2013-08-19
6	6	Santhosh	Kumar	santhosh@gmail.com	Testing	30000	2019-04-09
7	7	Lucida	Jose	jose@gmail.com	Marketing	30500	2009-02-24
8	8	Evan	Joseph	evan@gmail.com	Finance	32000	2014-07-09
9	9	Mukesh	Kumar	mukesh@gmail.com	Technical	40000	2020-03-05
10	10	Benjamin	Frankilin	frank@gmail.com	IT support	20000	2016-04-12

Task-1

1. Write an SQL query to fetch “FIRST_NAME” from the Worker table using the alias name as <WORKER_NAME>

```
7 •  SELECT first_name AS worker_name FROM Employee;
```

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

#	worker_name
1	Selvi
2	Ram
3	Alex
4	Alford
5	Ravi
6	Santhosh
7	Lucida
8	Evan
9	Mukesh
10	Benjamin
11	Tony
12	Tim
13	Kim
14	Sam
15	Kevin

2. Write an SQL query to fetch unique values of DEPARTMENT from the Worker table.

```
9 •  SELECT DISTINCT department FROM Employee;
```

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

#	department
1	Technical
2	Finance
3	Testing
4	Development
5	Marketing
6	IT support
7	Salesman
8	Manager
9	Analyst
10	Engineer

3. Write an SQL query to show the last 5 records from a table.

```
11    # Type 1
12 •  (SELECT * FROM Employee ORDER BY emp_no DESC LIMIT 5)
13    ORDER BY emp_no ASC;
14    # Type 2
15 •  SELECT * FROM Employee WHERE emp_no > (SELECT count(*) FROM Employee) - 5
```

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

#	emp_no	first_name	last_name	email	department	salary	join_date
1	26	Madii	Himbury	madii@gmail.com	Technical	30500	2005-02-16
2	27	Athena	Wilson	athena@yahoo.com	Analyst	32000	2015-04-10
3	28	Jennifer	Huette	jeni@guvi.com	Manager	80000	2021-04-15
4	29	John	Smith	josmi@yahoo.com	Engineer	60000	2009-04-12
5	30	Alan	Mathew	mathew@gmail.com	IT support	20000	2017-06-12

Task-2

1. Write an SQL query to print the first three characters of FIRST_NAME from Worker

```
17 •  SELECT LEFT(first_name, 3) FROM Employee;
```

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

#	LEFT(first_name, 3)
1	Sel
2	Ram
3	Ale
4	Alf
5	Rav
6	San
7	Luc
8	Eva
9	Muk
10	Ben

2. Write an SQL query to find the position of the alphabet (‘a’) in the first name

```
21 •  SELECT first_name, POSITION("a" in first_name) AS a_position FROM Employee;
```

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

#	first_name	a_position
1	Selvi	0
2	Ram	2
3	Alex	1
4	Alford	1
5	Ravi	2
6	Santhosh	2
7	Lucida	6
8	Evan	3
9	Mukesh	0
10	Benjamin	5
11	Tony	0

3. Write an SQL query to print the name of employees who have the highest salary in each department.

```
39 •  SELECT first_name, department, salary
40    FROM Employee
41    WHERE (department,salary) IN (
42        SELECT department, MAX(salary)
43        FROM Employee
44        GROUP BY department);
45
```

Result Grid | Filter Rows: | Export: | Wrap

#	first_name	department	salary
1	Lucida	Marketing	80500
2	Mukesh	Technical	85000
3	Benjamin	IT support	60000
4	Sam	Development	95000
5	Alfred	Testing	52000
6	Paul	Finance	90000
7	Kevin	Salesman	72500
8	Athena	Analyst	92000
9	Jennifer	Manager	80000
10	John	Engineer	99000

Task-3

1. Write an SQL query to print the FIRST_NAME from the Worker table after removing white spaces from the right side.

```
50 •  SELECT RTRIM(first_name) from Employee;
51
```

Result Grid | Filter Rows: | Export: | Wrap

#	RTRIM(first_name
1	Selvi
2	Ram
3	Alex
4	Alford
5	Ravi
6	Santhosh
7	Lucida
8	Evan
9	Mukesh
10	Benjamin

2. Write an SQL query that fetches the unique values of DEPARTMENT from the Worker table and prints its length.

```
54 •  SELECT department, LENGTH(department) AS length from Employee GROUP BY department;
```

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

#	department	length
1	Technical	9
2	Finance	7
3	Testing	7
4	Development	11
5	Marketing	9
6	IT support	10
7	Salesman	8
8	Manager	7
9	Analyst	7
10	Engineer	8

3. Write an SQL query to fetch n max salaries from a table.

```
92 •  SELECT first_name, salary FROM Employee ORDER BY salary DESC LIMIT 5;
```

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

#	first_name	salary
1	John	99000
2	Sam	95000
3	Athena	92000
4	Paul	90000
5	Mukesh	85000

Task-4

1. Write an SQL query to print the FIRST_NAME from the Worker table after replacing ‘a’ with ‘A’.

```
93 •  SELECT first_name, REPLACE(first_name, "a", "A") AS a_to_A FROM Employee;
94
```

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

#	first_name	a_to_A
1	Selvi	Selvi
2	Ram	RAm
3	Alex	Alex
4	Alford	Alford
5	Ravi	RAvi
6	Santhosh	SAnthosh
7	Lucida	LucidA
8	Evan	EvAn
9	Mukesh	Mukesh
10	Benjamin	BenjAmin
11	Tony	Tony
12	Tim	Tim
13	Kim	Kim
14	Sam	SAm

2. Write an SQL query to print all Worker details from the Worker table order FIRST_NAME Ascending and DEPARTMENT Descending.

```
93 •  SELECT *
94    FROM Employee
95    ORDER by first_name ASC, department DESC;
96
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
1	30	Alan	Mathew	mathew@gmail.com	IT support	20000	2017-06-12
2	3	Alex	George	alexgeorge@guvi.com	Finance	20000	2016-11-12
3	4	Alford	Francis	alfordfrancis@guvi.com	Testing	25000	2010-07-09
4	17	Alfred	Kinsley	kinsley@gmail.com	Testing	52000	2017-07-09
5	21	Andrew	Faulkner	andrew@guvi.com	Development	55000	2016-08-05
6	27	Athena	Wilson	athena@yahoo.com	Analyst	92000	2015-04-10
7	24	Bella	Swan	bella@solutions.in	Development	55000	2013-08-09
8	10	Benjamin	Frankilin	frank@gmail.com	IT support	60000	2016-04-12
9	16	Connie	Smith	smith@gmail.com	Marketing	30500	2009-02-14
10	8	Evan	Joseph	evanjoseph@guvi.com	Finance	32000	2014-07-09
11	28	Jennifer	Huette	jeni@guvi.com	Manager	80000	2021-04-15
12	29	John	Smith	johnsmith@yahoo.com	Engineer	99000	2009-04-12
13	19	John	Asghar	john@yahoo.com	Engineer	60000	2019-04-12

3. Write an SQL query to fetch the names of workers who earn the highest salary.

```
93 •  SELECT first_name, salary FROM Employee ORDER BY salary DESC LIMIT 1;
94
```

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

#	first_name	salary
1	John	99000

Task-5

1. Write an SQL query to print details of workers excluding first names, “Rose” and “Paul” from the Worker table.

```
92 •  SELECT * FROM Employee WHERE first_name  NOT IN ('Rose', 'Paul');
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
11	11	Tony	Stark	stark@guvi.com	Salesman	35000	2013-08-05
12	12	Tim	Adolf	adolf@guvi.com	Manager	50000	2014-11-12
13	13	Kim	Jarvis	kim@gmail.com	Testing	35000	2017-08-09
14	14	Sam	Miles	sammiles@solutions.in	Development	95000	2018-08-19
15	15	Kevin	Hill	kevinhill@guvi.com	Analyst	60000	2019-04-09
16	16	Connie	Smith	smith@gmail.com	Marketing	30500	2009-02-14
17	17	Alfred	Kinsley	kinsley@gmail.com	Testing	52000	2017-07-09
18	19	John	Asghar	john@yahoo.com	Engineer	60000	2019-04-12
19	21	Andrew	Faulkner	andrew@guvi.com	Development	55000	2016-08-05
20	22	Karen	Mathew	karen@guvi.com	Salesman	40000	2017-11-12
21	23	Wendy	Shawn	shawn@guvi.com	Testing	35000	2019-08-10
22	24	Bella	Swan	bella@solutions.in	Development	55000	2013-08-09
23	25	Kevin	Hill	hillkevin@guvi.com	Salesman	72500	2019-07-09
24	26	Madii	Himbury	madii@gmail.com	Technical	30500	2005-02-16
25	27	Athena	Wilson	athena@yahoo.com	Analyst	92000	2015-04-10
26	28	Jennifer	Huette	jeni@guvi.com	Manager	80000	2021-04-15
27	29	John	Smith	johnsmith@yahoo.com	Engineer	99000	2009-04-12
28	30	Alan	Mathew	mathew@gmail.com	IT support	20000	2017-06-12
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2. Write an SQL query to print details of the Workers whose FIRST_NAME ends with ‘h’ and contains six alphabets.

```
92 •  SELECT *
93    FROM Employee
94    WHERE (first_name REGEXP 'h$') AND (LENGTH(first_name)=6);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
1	9	Mukesh	Kumar	mukeshkumar@guvi.com	Technical	85000	2020-03-05
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

3. Write a query to validate Email of Employee (email should have first name last name and guvi.com example (first name=Kamal last name= raja and the mail id should be kamalraja@guvi.com).

```
89 •  SELECT * FROM Employee
90    WHERE (email REGEXP CONCAT(LOWER(RTRIM(first_name)),LOWER(last_name), '@guvi.com'));
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
1	1	Selvi	Jayaraman	selvijayaraman@guvi.com	Technical	30000	2019-07-09
2	3	Alex	George	alexgeorge@guvi.com	Finance	20000	2016-11-12
3	4	Alford	Francis	alfordfrancis@guvi.com	Testing	25000	2010-07-09
4	6	Santhosh	Kumar	santhoshkumar@guvi.com	Testing	30000	2019-04-09
5	8	Evan	Joseph	evanjoseph@guvi.com	Finance	32000	2014-07-09
6	9	Mukesh	Kumar	mukeshkumar@guvi.com	Technical	85000	2020-03-05
7	15	Kevin	Hill	kevinhill@guvi.com	Analyst	60000	2019-04-09
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Note: Added some duplicate rows in the existing table

Task-6

1. Write an SQL query to print details of the Workers who have joined in March ’2021.

```
94 •  SELECT first_name, join_date
95    FROM Employee
96    WHERE EXTRACT(YEAR_MONTH FROM join_date)=202104;
```

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

#	first_name	join_date
1	Jennifer	2021-04-15

2. Write an SQL query to fetch duplicates that have matching data in some fields of a table.

```sql
SELECT a.*
FROM Employee a
JOIN (SELECT email, COUNT(*) as frequency
FROM Employee
GROUP BY email
HAVING count(*) > 1) b
ON a.email = b.email
ORDER BY a.email;
```

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

#	emp_no	first_name	last_name	email	department	salary	join_date
1	24	Andrew	Faulkner	andrew@guvi.com	Development	55000	2016-08-05
2	4	Andrew	Faulkner	andrew@guvi.com	Development	55000	2016-08-05
3	9	Evan	Joseph	evanjoseph@guvi.com	Finance	32000	2014-07-09
4	16	Evan	Joseph	evanjoseph@guvi.com	Finance	32000	2014-07-09
5	11	John	Smith	johnsmith@yahoo.com	Engineer	99000	2009-04-12
6	34	John	Smith	johnsmith@yahoo.com	Engineer	99000	2009-04-12
7	23	Rose	Summers	rose@guvi.com	IT support	20000	2007-06-12
8	32	Rose	Summers	rose@guvi.com	IT support	20000	2007-06-12
9	17	Sam	Miles	sammiles@solutions.in	Development	95000	2018-08-19
10	31	Sam	Miles	sammiles@solutions.in	Development	95000	2018-08-19
11	35	Sam	Miles	sammiles@solutions.in	Development	95000	2018-08-19

3. How to remove duplicate rows from the Employees table.

   Note: Removed 6 duplicate records

```sql
# How to remove duplicate rows from the Employees table.
# Step1 - To fetch duplicate rows
SELECT *
    FROM Employee
    WHERE emp_no NOT IN
    (
        SELECT MAX(emp_no)
        FROM Employee
        GROUP BY email
    );
# Step2 - To delete duplicate rows
DELETE FROM Employee
    WHERE emp_no NOT IN
    (
        SELECT MAX(emp_no)
        FROM (SELECT * FROM Employee) AS deleteRecord
        GROUP BY email
    );
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
26	32	Rose	Summers	rose@guvi.com	IT support	20000	2007-06-12
27	33	Jennifer	Huette	jeni@guvi.com	Manager	80000	2021-04-15
28	34	John	Smith	johnsmith@yahoo.com	Engineer	99000	2009-04-12
29	35	Sam	Miles	sammiles@solutions.in	Development	95000	2018-08-19
30	36	Alan	Mathew	mathew@gmail.com	IT support	20000	2017-06-12
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Task-7

1. Write an SQL query to show only odd rows from a table.

```sql
# Write an SQL query to show only odd rows from a table.
SELECT *
FROM Employee
WHERE emp_no NOT IN
    (
        SELECT emp_no
        FROM Employee
        WHERE MOD(emp_no,2)=0
    );
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
1	1	Selvi	Jayaraman	selvijayaraman@guvi.com	Technical	30000	2019-07-09
2	3	Alex	George	alexgeorge@guvi.com	Finance	20000	2016-11-12
3	5	Alford	Francis	alfordfrancis@guvi.com	Testing	25000	2010-07-09
4	7	Santhosh	Kumar	santhoshkumar@guvi.com	Testing	30000	2019-04-09
5	9	Evan	Joseph	evanjoseph@guvi.com	Finance	32000	2014-07-09
6	11	John	Smith	johnsmith@yahoo.com	Engineer	99000	2009-04-12
7	13	Tony	Stark	stark@guvi.com	Salesman	35000	2013-08-05
8	15	Kim	Jarvis	kim@gmail.com	Testing	35000	2017-08-09
9	17	Sam	Miles	sammiles@solutions.in	Development	95000	2018-08-19
10	19	Connie	Smith	smith@gmail.com	Marketing	30500	2009-02-14
11	21	Paul	Timothy	paultimothyh@guvi.com	Finance	90000	2020-03-15
12	23	Rose	Summers	rose@guvi.com	IT support	20000	2007-06-12
13	25	Karen	Mathew	karen@guvi.com	Salesman	40000	2017-11-12
14	27	Bella	Swan	bella@solutions.in	Development	55000	2013-08-09

2. Write an SQL query to clone a new table from another table.

```sql
CREATE TABLE clone_employee LIKE Employee;
INSERT INTO clone_employee SELECT * FROM Employee;
SHOW TABLES;
DESCRIBE clone_employee;
SELECT * FROM clone_employee;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap

#	emp_no	first_name	last_name	email	department	salary	join_date
1	1	Selvi	Jayaraman	selvijayaraman@guvi.com	Technical	30000	2019-07-09
2	2	Ram	Sundar	ram@gmail.com	Technical	35000	2018-08-05
3	3	Alex	George	alexgeorge@guvi.com	Finance	20000	2016-11-12
4	4	Andrew	Faulkner	andrew@guvi.com	Development	55000	2016-08-05
5	5	Alford	Francis	alfordfrancis@guvi.com	Testing	25000	2010-07-09
6	6	Ravi	Kumar	kumar@solutions.in	Development	40000	2013-08-19
7	7	Santhosh	Kumar	santhoshkumar@guvi.com	Testing	30000	2019-04-09
8	8	Lucida	Jose	lucidajose@gmail.com	Marketing	80500	2009-02-24
9	9	Evan	Joseph	evanjoseph@guvi.com	Finance	32000	2014-07-09

Task-8

1. Write an SQL query to fetch intersecting records of two tables.
   Created new table called “Experience”

```
222 •  SELECT * FROM Experience;
```

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

#	emp_no	email	technology	years_of_experience
1	2	ram@gmail.com	Front-end	3
2	3	alexgeorge@guvi.com	Accounts	4
3	5	alfordfrancis@guvi.com	Selenium	5
4	6	kumar@solutions.in	Back-end	6
5	27	bella@solutions.in	Back-end	2
6	28	hillkevin@guvi.com	BI tools	3
7	29	madii@gmail.com	Front-end	7
8	30	athena@yahoo.com	BI tools	5
9	31	sammiles@solutions.in	Back-end	3
10	32	rose@guvi.com	OS	4
11	33	jeni@guvi.com	BI tools	6
12	7	santhoshkumar@guv…	Selenium	2

Fetching intersecting records

```
224 •  SELECT Employee.email, Employee.department, Experience.technology, Experience.years_of_experience
225    FROM Employee
226    INNER JOIN Experience ON Employee.emp_no = Experience.emp_no;
```

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

#	email	department	technology	years_of_experience
1	ram@gmail.com	Technical	Front-end	3
2	alexgeorge@guvi.com	Finance	Accounts	4
3	alfordfrancis@guvi.com	Testing	Selenium	5
4	kumar@solutions.in	Development	Back-end	6
5	bella@solutions.in	Development	Back-end	2
6	hillkevin@guvi.com	Salesman	BI tools	3
7	madii@gmail.com	Technical	Front-end	7
8	athena@yahoo.com	Analyst	BI tools	5
9	sammiles@solutions.in	Development	Back-end	3
10	rose@guvi.com	IT support	OS	4
11	jeni@guvi.com	Manager	BI tools	6
12	santhoshkumar@guvi.com	Testing	Selenium	2
13	mukeshkumar@guvi.com	Technical	Front-end	6

2. Write an SQL query to show records from one table that another table does not have.

```
235 •  SELECT Employee.email, Experience.technology, Experience.years_of_experience
236    FROM Employee
237    RIGHT JOIN Experience ON Employee.emp_no = Experience.emp_no
238    ORDER BY years_of_experience;
```

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

#	email	technology	years_of_experience
10	alexgeorge@guvi.com	Accounts	4
11	rose@guvi.com	OS	4
12	alfordfrancis@guvi.com	Selenium	5
13	athena@yahoo.com	BI tools	5
14	NULL	Back-end	5
15	jeni@guvi.com	BI tools	6
16	mukeshkumar@guvi.com	Front-end	6
17	kumar@solutions.in	Back-end	6
18	kim@gmail.com	Selenium	7
19	NULL	Editing	7
20	madii@gmail.com	Front-end	7
21	johnsmith@yahoo.com	Back-end	8
22	NULL	Back-end	8

Task-9

1. Write an SQL query to show the top n (say 15) records of a table.

```
240 •  SELECT *
241    FROM Employee
242    LIMIT 15;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch r

#	emp_no	first_name	last_name	email	department	salary	join_date
5	5	Alford	Francis	alfordfrancis@guvi.com	Testing	25000	2010-07-09
6	6	Ravi	Kumar	kumar@solutions.in	Development	40000	2013-08-19
7	7	Santhosh	Kumar	santhoshkumar@guvi.com	Testing	30000	2019-04-09
8	8	Lucida	Jose	lucidajose@gmail.com	Marketing	80500	2009-02-24
9	9	Evan	Joseph	evanjoseph@guvi.com	Finance	32000	2014-07-09
10	10	Mukesh	Kumar	mukeshkumar@guvi.com	Technical	85000	2020-03-05
11	11	John	Smith	johnsmith@yahoo.com	Engineer	99000	2009-04-12
12	12	Benjamin	Frankilin	frank@gmail.com	IT support	60000	2016-04-12
13	13	Tony	Stark	stark@guvi.com	Salesman	35000	2013-08-05
14	14	Tim	Adolf	adolf@guvi.com	Manager	50000	2014-11-12
15	15	Kim	Jarvis	kim@gmail.com	Testing	35000	2017-08-09
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2. Write an SQL query to determine the nth (say n=15) highest salary from a table.

```
259 •  SELECT * FROM Employee ORDER BY salary DESC LIMIT 14,1;
260
```

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

#	emp_no	first_name	last_name	email	department	salary	join_date
1	27	Bella	Swan	bella@solutions.in	Development	55000	2013-08-09
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Task-10

1. Write an SQL query to determine the 8th highest salary without using TOP or LIMIT methods.

```
284 •  SELECT *
285    FROM
286        ( SELECT ROW_NUMBER()
287          OVER (ORDER BY SALARY DESC) AS rownumber,Salary
288          FROM Employee
289        )
290    AS salary
291    WHERE rownumber = 8;
```

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

#	rownumber	Salary
1	8	85000

2. Write an SQL query to fetch the list of employees with the same salary.

```
295 •  SELECT * FROM Employee
296    WHERE SALARY IN(
297        SELECT salary FROM Employee
298        GROUP BY salary
299        HAVING COUNT(emp_no)>1)
300    ORDER BY salary DESC;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

#	emp_no	first_name	last_name	email	department	salary	join_date
1	12	Benjamin	Frankilin	frank@gmail.com	IT support	60000	2016-04-12
2	18	Kevin	Hill	kevinhill@guvi.com	Analyst	60000	2019-04-09
3	22	John	Asghar	john@yahoo.com	Engineer	60000	2019-04-12
4	27	Bella	Swan	bella@solutions.in	Development	55000	2013-08-09
5	24	Andrew	Faulkner	andrew@guvi.com	Development	55000	2016-08-05
6	6	Ravi	Kumar	kumar@solutions.in	Development	40000	2013-08-19
7	25	Karen	Mathew	karen@guvi.com	Salesman	40000	2017-11-12
8	38	Tony	Stark	stark@guvi.com	Salesman	35000	2013-08-05
9	26	Wendy	Shawn	shawn@guvi.com	Testing	35000	2019-08-10
10	15	Kim	Jarvis	kim@gmail.com	Testing	35000	2017-08-09
11	2	Ram	Sundar	ram@gmail.com	Technical	35000	2018-08-05
12	19	Connie	Smith	smith@gmail.com	Marketing	30500	2009-02-14
13	29	Madii	Himbury	madii@gmail.com	Technical	30500	2005-02-16
14	7	Santhosh	Kumar	santhoshkumar@guvi.com	Testing	30000	2019-04-09
15	1	Selvi	Jayaraman	selvijayaraman@guvi.com	Technical	30000	2019-07-09