Name: Krushnakumar Patle

Email: krishnapatle128@gmail.com

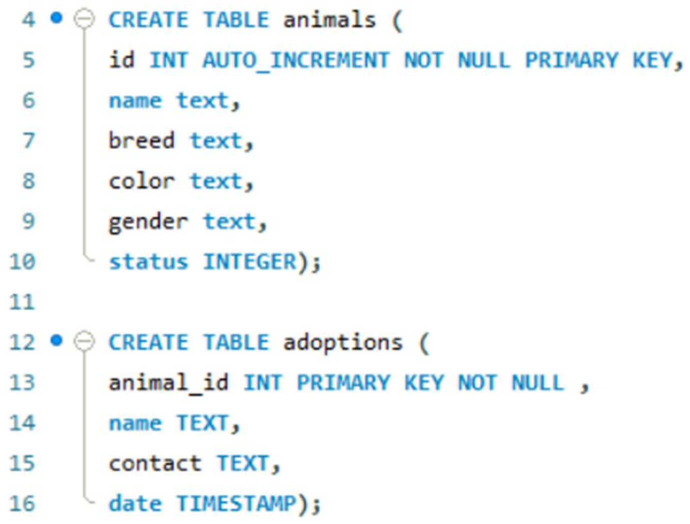
Batch: Data Engineering Batch-1

1. Create database pet_adoption and use it

   create database pet_adoption;

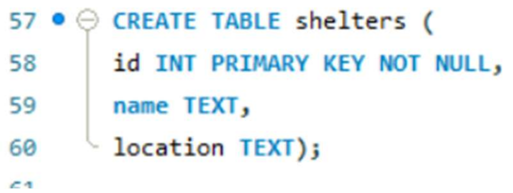
   use pet_adoption;

2. Create animal and adoption table



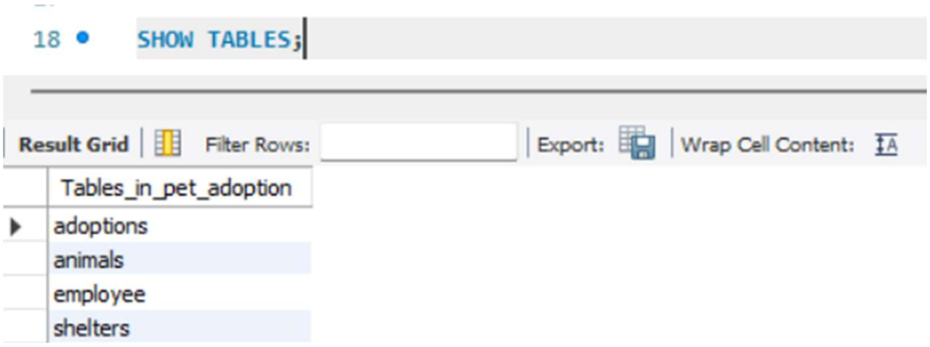
```
CREATE TABLE animals (
id INT AUTO_INCREMENT NOT NULL PRIMARY KEY,
name text,
breed text,
color text,
gender text,
status INTEGER);

CREATE TABLE adoptions (
animal_id INT PRIMARY KEY NOT NULL ,
name TEXT,
contact TEXT,
date TIMESTAMP);
```



```
CREATE TABLE shelters (
id INT PRIMARY KEY NOT NULL,
name TEXT,
location TEXT);
```

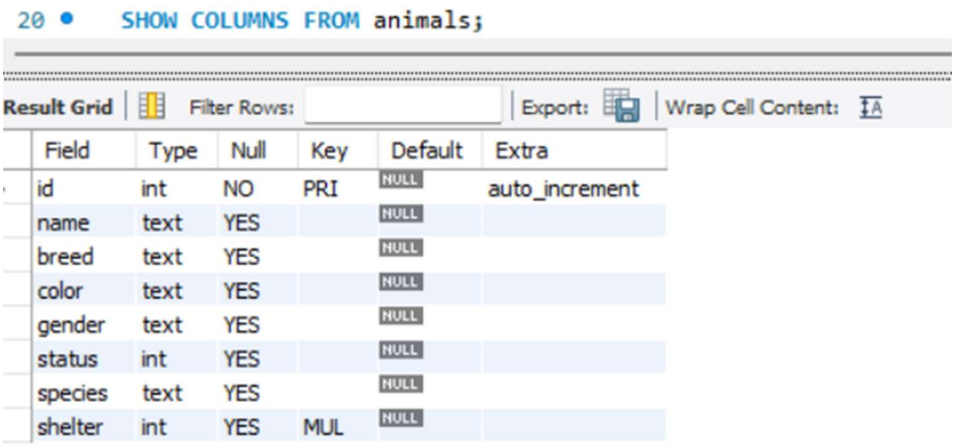
3. Show tables.



```
SHOW TABLES;
```

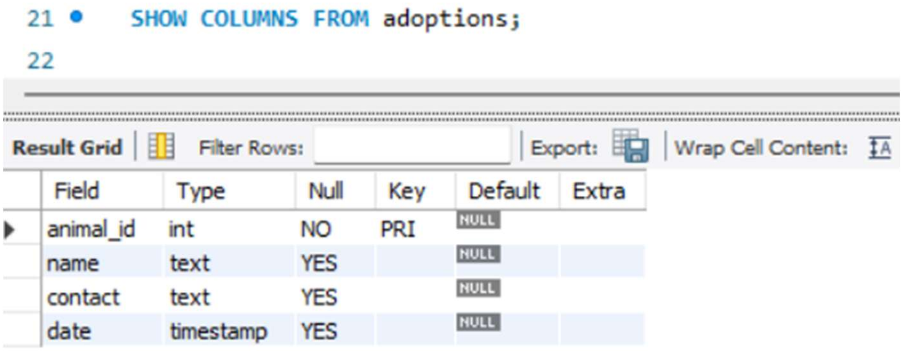
Tables_in_pet_adoption
adoptions
animals
employee
shelters

4. Show columns



```
SHOW COLUMNS FROM animals;
```

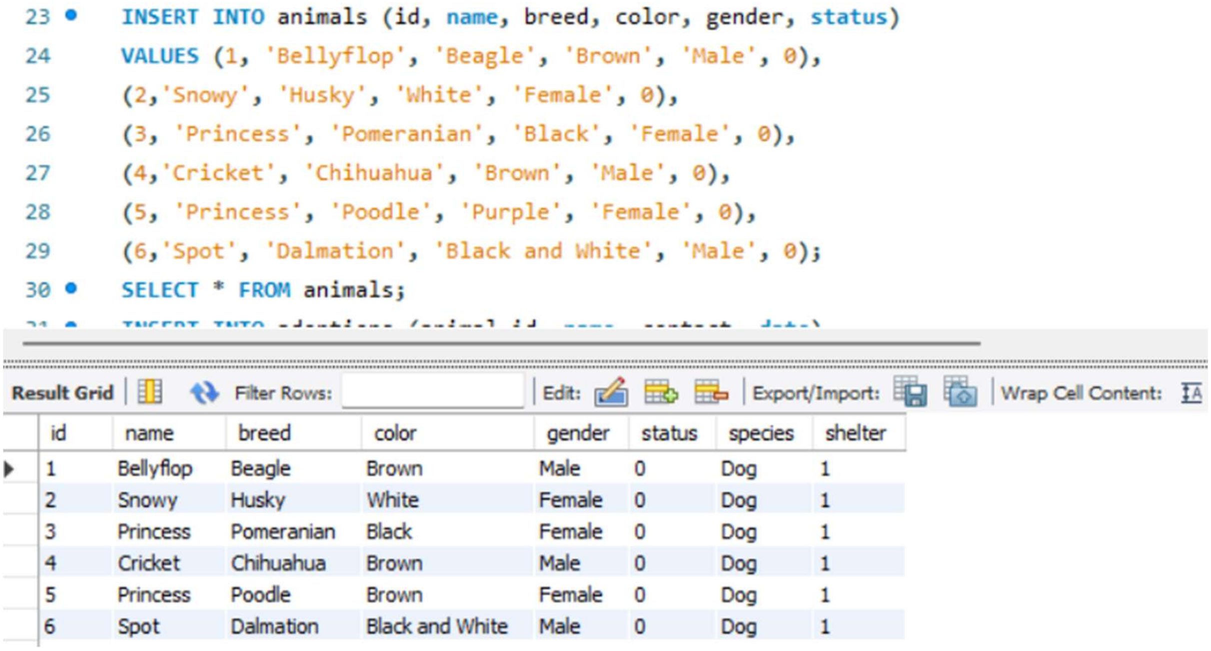
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
name	text	YES		NULL	
breed	text	YES		NULL	
color	text	YES		NULL	
gender	text	YES		NULL	
status	int	YES		NULL	
species	text	YES		NULL	
shelter	int	YES	MUL	NULL	



```
SHOW COLUMNS FROM adoptions;
```

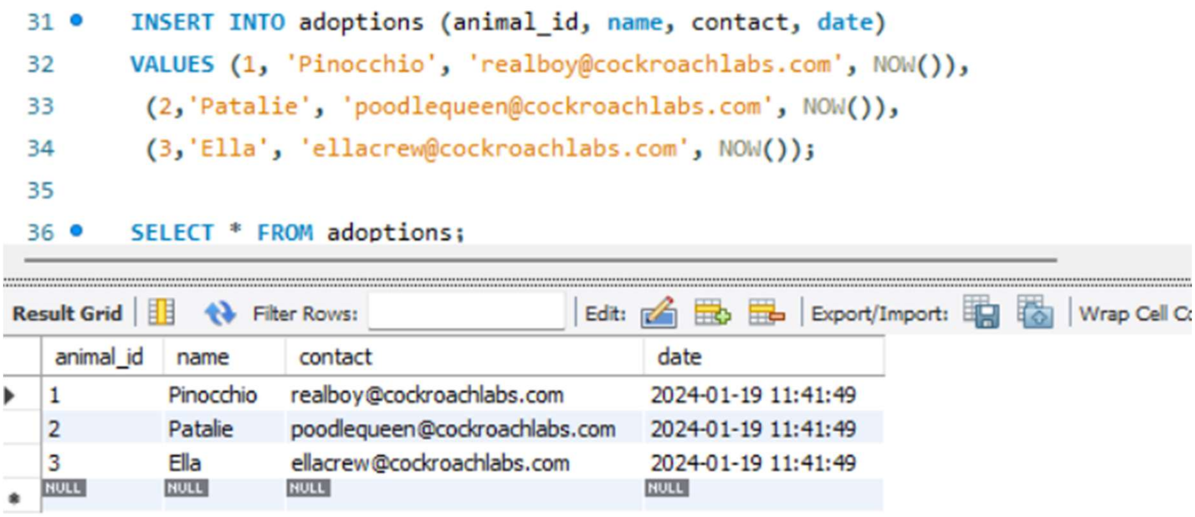
Field	Type	Null	Key	Default	Extra
animal_id	int	NO	PRI	NULL	
name	text	YES		NULL	
contact	text	YES		NULL	
date	timestamp	YES		NULL	

5. Insert data into tables



```
INSERT INTO animals (id, name, breed, color, gender, status)
VALUES (1, 'Bellyflop', 'Beagle', 'Brown', 'Male', 0),
(2,'Snowy', 'Husky', 'White', 'Female', 0),
(3, 'Princess', 'Pomeranian', 'Black', 'Female', 0),
(4,'Cricket', 'Chihuahua', 'Brown', 'Male', 0),
(5, 'Princess', 'Poodle', 'Purple', 'Female', 0),
(6,'Spot', 'Dalmation', 'Black and White', 'Male', 0);
SELECT * FROM animals;
INSERT INTO adoptions (animal_id, name, contact, date)
```

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
2	Snowy	Husky	White	Female	0	Dog	1
3	Princess	Pomeranian	Black	Female	0	Dog	1
4	Cricket	Chihuahua	Brown	Male	0	Dog	1
5	Princess	Poodle	Brown	Female	0	Dog	1
6	Spot	Dalmation	Black and White	Male	0	Dog	1

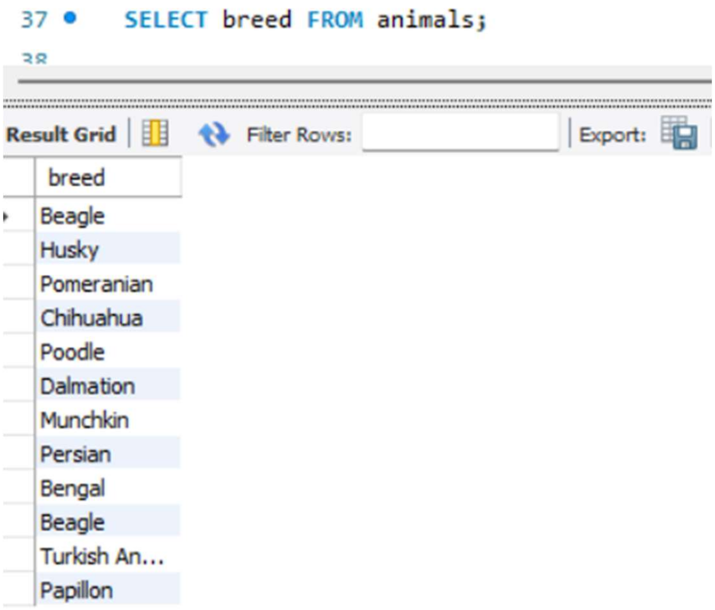


```sql
INSERT INTO adoptions (animal_id, name, contact, date)
VALUES (1, 'Pinocchio', 'realboy@cockroachlabs.com', NOW()),
 (2,'Patalie', 'poodlequeen@cockroachlabs.com', NOW()),
 (3,'Ella', 'ellacrew@cockroachlabs.com', NOW());

SELECT * FROM adoptions;
```

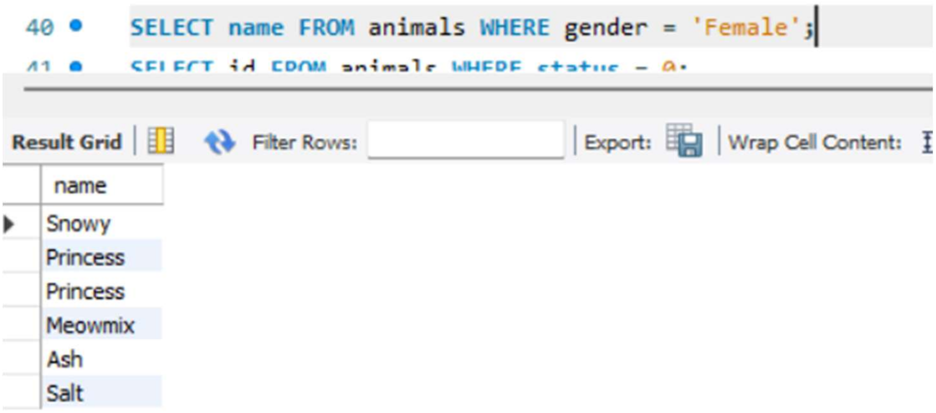
animal_id	name	contact	date
1	Pinocchio	realboy@cockroachlabs.com	2024-01-19 11:41:49
2	Patalie	poodlequeen@cockroachlabs.com	2024-01-19 11:41:49
3	Ella	ellacrew@cockroachlabs.com	2024-01-19 11:41:49
NULL	NULL	NULL	NULL

6. SELECT Command



```sql
SELECT breed FROM animals;
```

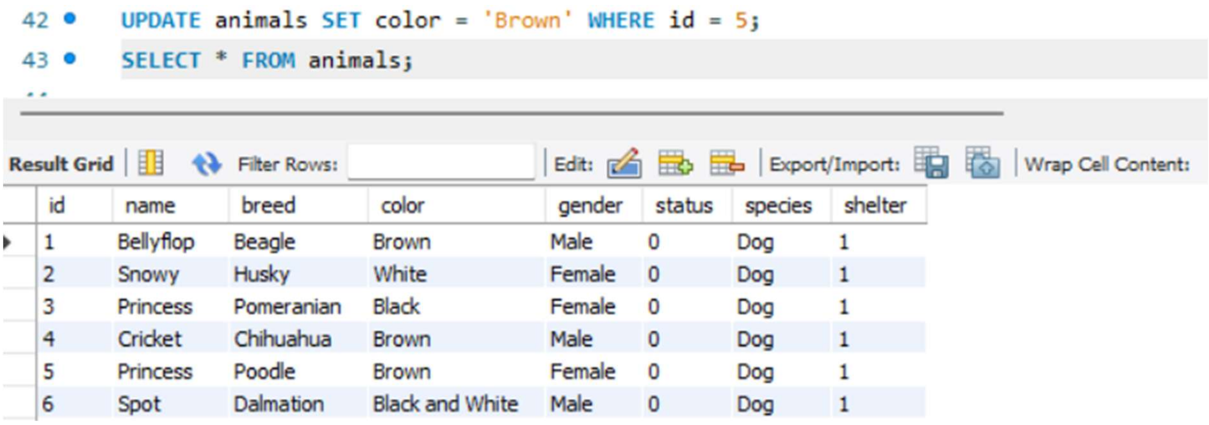
breed
Beagle
Husky
Pomeranian
Chihuahua
Poodle
Dalmation
Munchkin
Persian
Bengal
Beagle
Turkish An...
Papillon



```sql
SELECT name FROM animals WHERE gender = 'Female';
SELECT id FROM animals WHERE status = 0;
```

name
Snowy
Princess
Princess
Meowmix
Ash
Salt

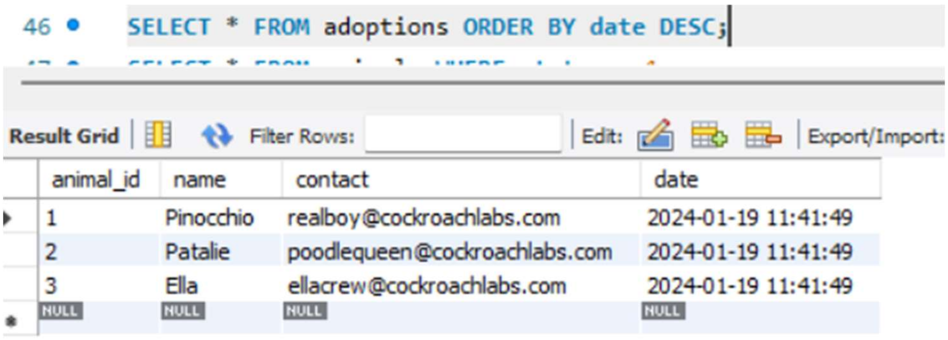
7. UPDATE Command



```sql
UPDATE animals SET color = 'Brown' WHERE id = 5;
SELECT * FROM animals;
```

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
2	Snowy	Husky	White	Female	0	Dog	1
3	Princess	Pomeranian	Black	Female	0	Dog	1
4	Cricket	Chihuahua	Brown	Male	0	Dog	1
5	Princess	Poodle	Brown	Female	0	Dog	1
6	Spot	Dalmation	Black and White	Male	0	Dog	1

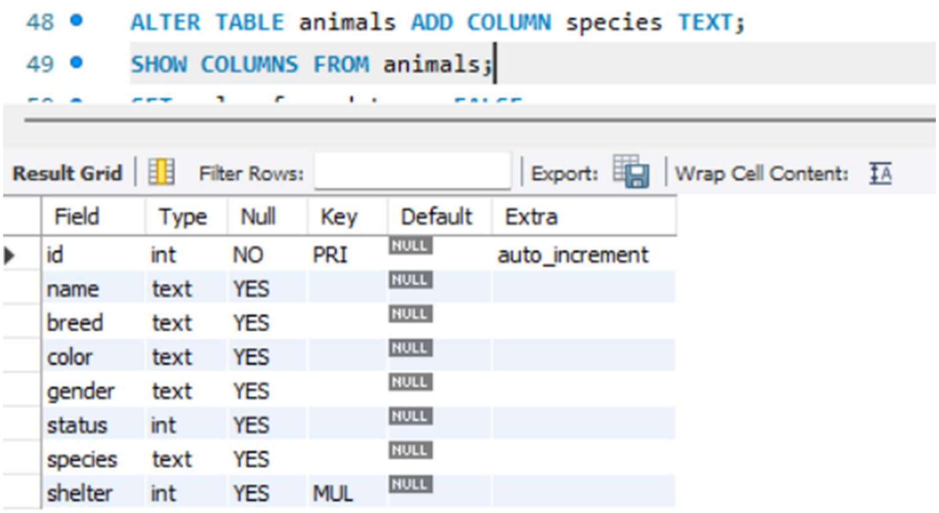
8. ORDER BY Command



```sql
SELECT * FROM adoptions ORDER BY date DESC;
```

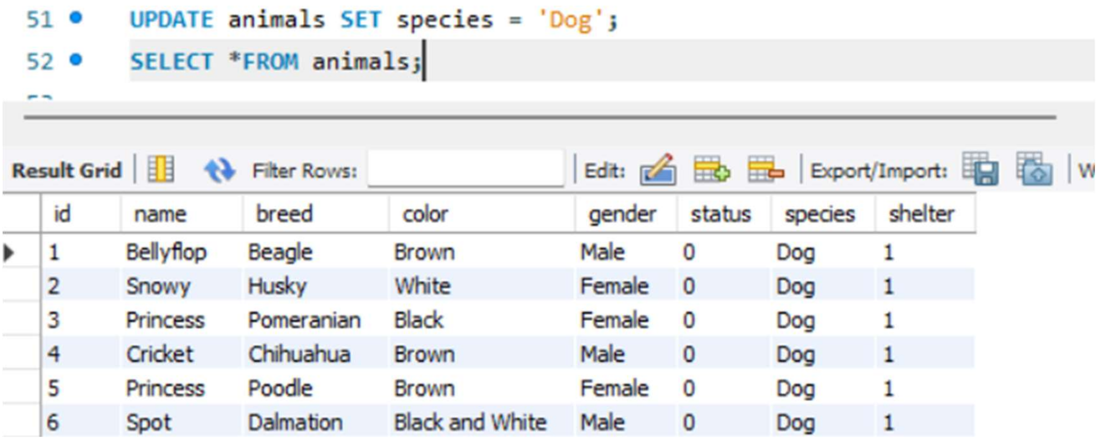
animal_id	name	contact	date
1	Pinocchio	realboy@cockroachlabs.com	2024-01-19 11:41:49
2	Patalie	poodlequeen@cockroachlabs.com	2024-01-19 11:41:49
3	Ella	ellacrew@cockroachlabs.com	2024-01-19 11:41:49
NULL	NULL	NULL	NULL

9. ALTER Command



```sql
ALTER TABLE animals ADD COLUMN species TEXT;
SHOW COLUMNS FROM animals;
```

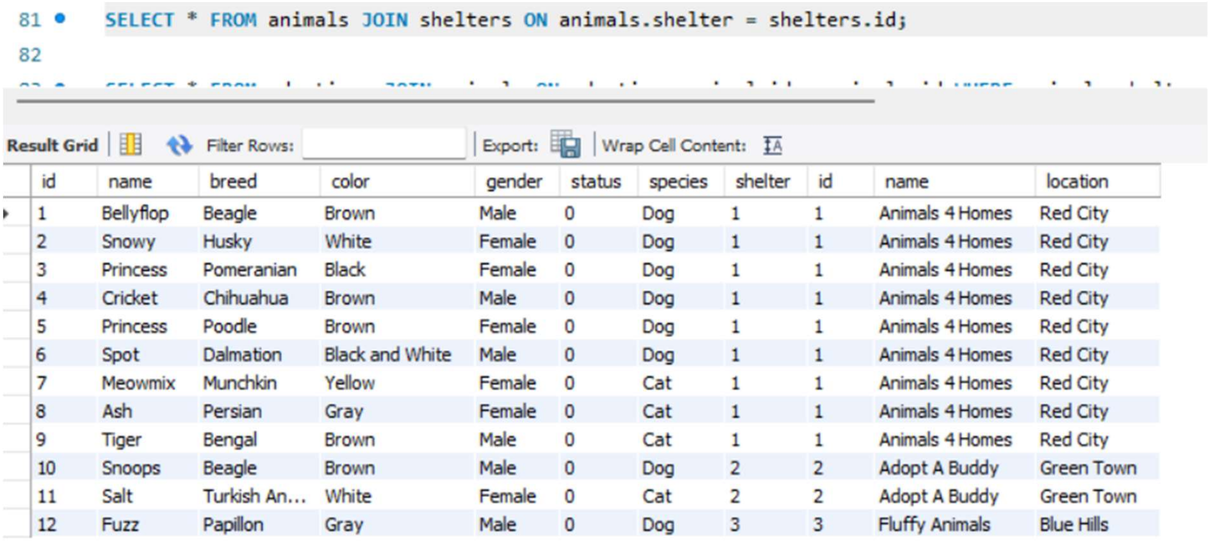
Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
name	text	YES		NULL	
breed	text	YES		NULL	
color	text	YES		NULL	
gender	text	YES		NULL	
status	int	YES		NULL	
species	text	YES		NULL	
shelter	int	YES	MUL	NULL	



```sql
UPDATE animals SET species = 'Dog';
SELECT *FROM animals;
```

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
2	Snowy	Husky	White	Female	0	Dog	1
3	Princess	Pomeranian	Black	Female	0	Dog	1
4	Cricket	Chihuahua	Brown	Male	0	Dog	1
5	Princess	Poodle	Brown	Female	0	Dog	1
6	Spot	Dalmation	Black and White	Male	0	Dog	1

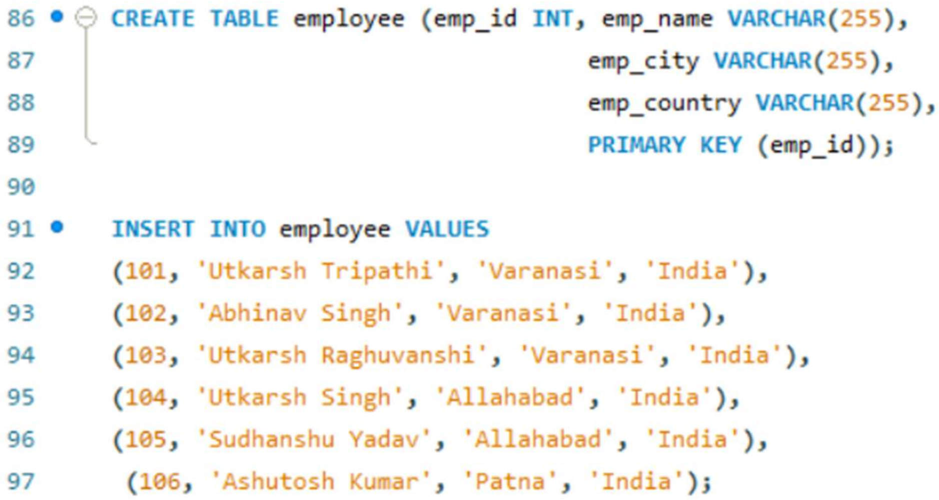
10.JOIN Command



```sql
SELECT * FROM animals JOIN shelters ON animals.shelter = shelters.id;
```

id	name	breed	color	gender	status	species	shelter	id	name	location
1	Bellyflop	Beagle	Brown	Male	0	Dog	1	1	Animals 4 Homes	Red City
2	Snowy	Husky	White	Female	0	Dog	1	1	Animals 4 Homes	Red City
3	Princess	Pomeranian	Black	Female	0	Dog	1	1	Animals 4 Homes	Red City
4	Cricket	Chihuahua	Brown	Male	0	Dog	1	1	Animals 4 Homes	Red City
5	Princess	Poodle	Brown	Female	0	Dog	1	1	Animals 4 Homes	Red City
6	Spot	Dalmation	Black and White	Male	0	Dog	1	1	Animals 4 Homes	Red City
7	Meowmix	Munchkin	Yellow	Female	0	Cat	1	1	Animals 4 Homes	Red City
8	Ash	Persian	Gray	Female	0	Cat	1	1	Animals 4 Homes	Red City
9	Tiger	Bengal	Brown	Male	0	Cat	1	1	Animals 4 Homes	Red City
10	Snoops	Beagle	Brown	Male	0	Dog	2	2	Adopt A Buddy	Green Town
11	Salt	Turkish An...	White	Female	0	Cat	2	2	Adopt A Buddy	Green Town
12	Fuzz	Papillon	Gray	Male	0	Dog	3	3	Fluffy Animals	Blue Hills

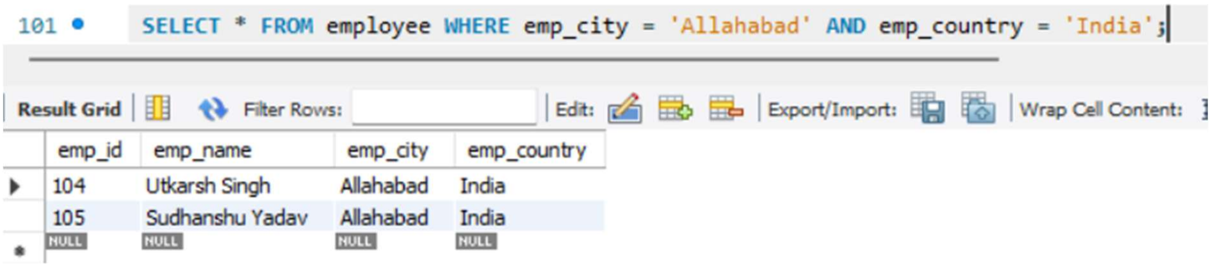
11.CREATE Command



```sql
CREATE TABLE employee (emp_id INT, emp_name VARCHAR(255),
                                   emp_city VARCHAR(255),
                                   emp_country VARCHAR(255),
                                   PRIMARY KEY (emp_id));

INSERT INTO employee VALUES
(101, 'Utkarsh Tripathi', 'Varanasi', 'India'),
(102, 'Abhinav Singh', 'Varanasi', 'India'),
(103, 'Utkarsh Raghuvanshi', 'Varanasi', 'India'),
(104, 'Utkarsh Singh', 'Allahabad', 'India'),
(105, 'Sudhanshu Yadav', 'Allahabad', 'India'),
 (106, 'Ashutosh Kumar', 'Patna', 'India');
```

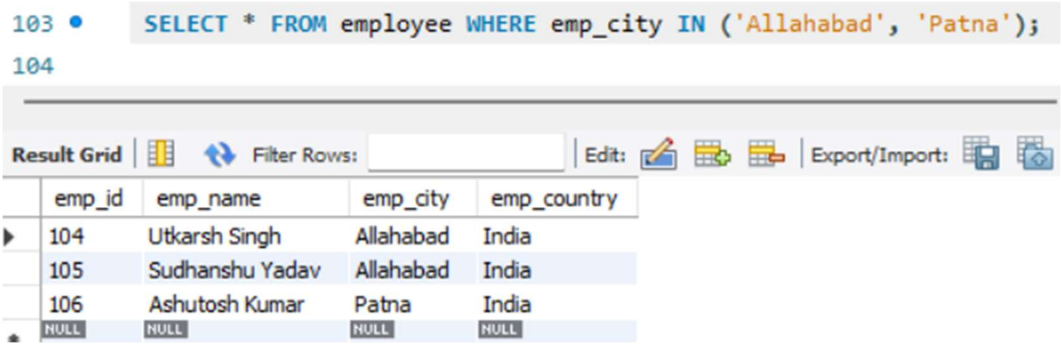
12.WHERE AND Command



```
SELECT * FROM employee WHERE emp_city = 'Allahabad' AND emp_country = 'India';
```

emp_id	emp_name	emp_city	emp_country
104	Utkarsh Singh	Allahabad	India
105	Sudhanshu Yadav	Allahabad	India
NULL	NULL	NULL	NULL

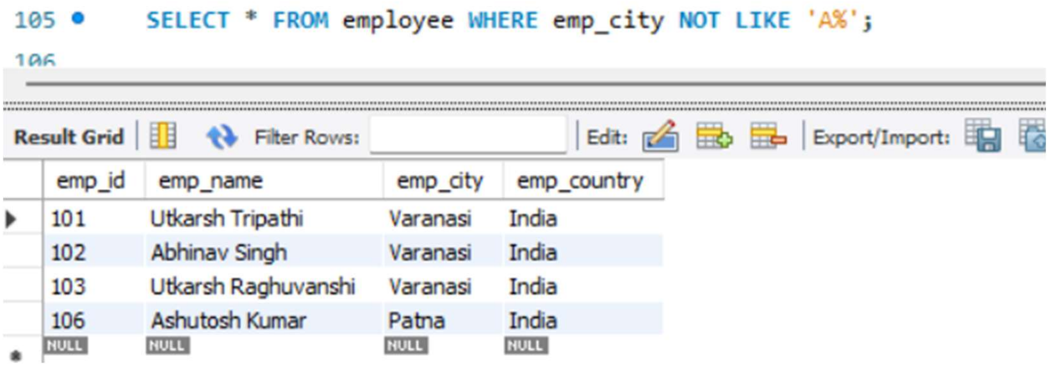
13.IN Command



```
SELECT * FROM employee WHERE emp_city IN ('Allahabad', 'Patna');
```

emp_id	emp_name	emp_city	emp_country
104	Utkarsh Singh	Allahabad	India
105	Sudhanshu Yadav	Allahabad	India
106	Ashutosh Kumar	Patna	India
NULL	NULL	NULL	NULL

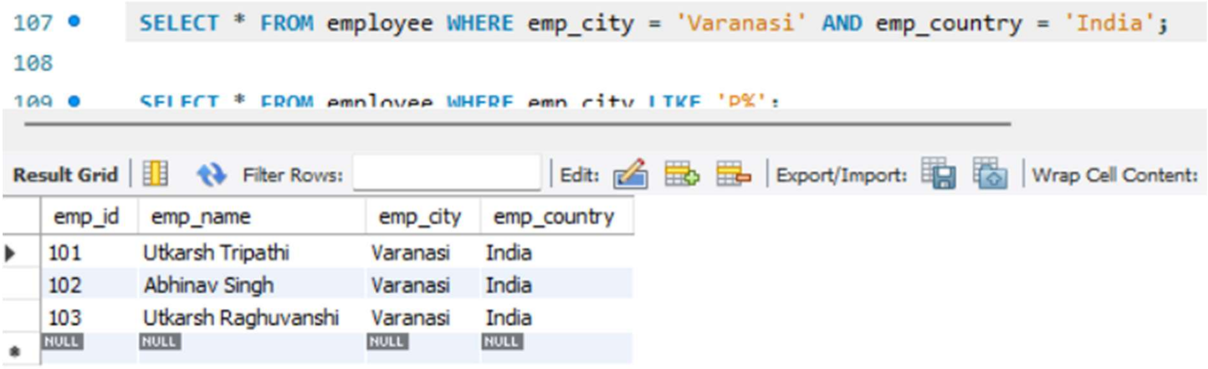
14.LIKE Command



```
SELECT * FROM employee WHERE emp_city NOT LIKE 'A%';
```

emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
106	Ashutosh Kumar	Patna	India
NULL	NULL	NULL	NULL

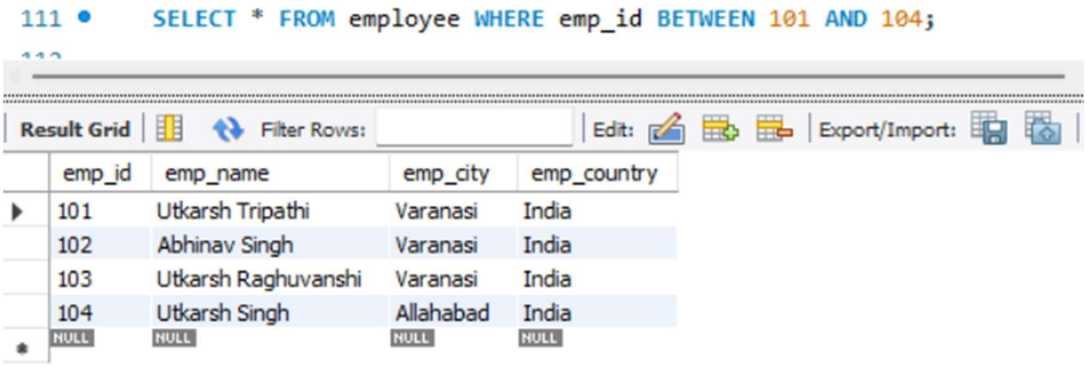
15.AND Command



```sql
SELECT * FROM employee WHERE emp_city = 'Varanasi' AND emp_country = 'India';

SELECT * FROM employee WHERE emp_city LIKE 'P%';
```

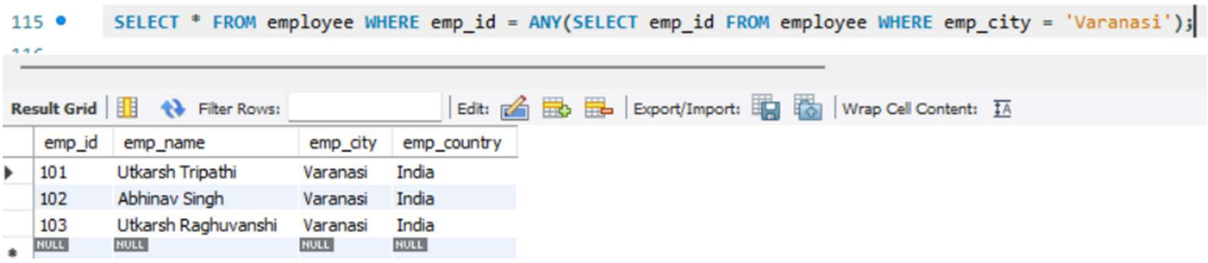
emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
NULL	NULL	NULL	NULL



```sql
SELECT * FROM employee WHERE emp_id BETWEEN 101 AND 104;
```

emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
104	Utkarsh Singh	Allahabad	India
NULL	NULL	NULL	NULL

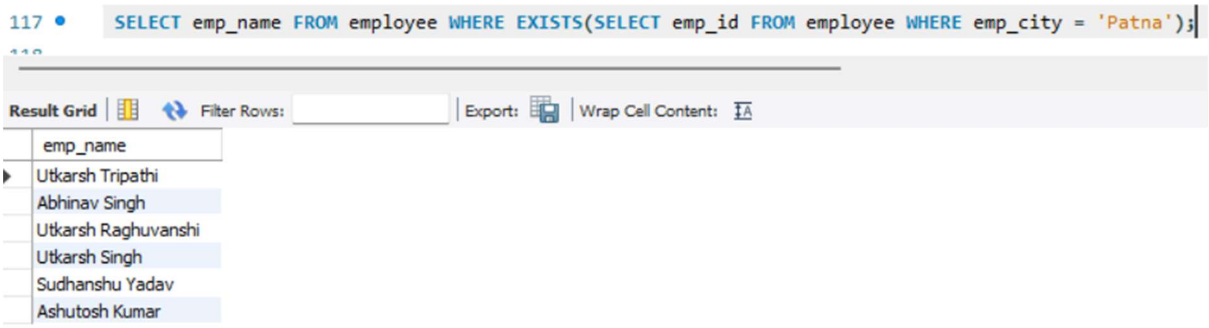
16.ANY Command



```sql
SELECT * FROM employee WHERE emp_id = ANY(SELECT emp_id FROM employee WHERE emp_city = 'Varanasi');
```

emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
NULL	NULL	NULL	NULL

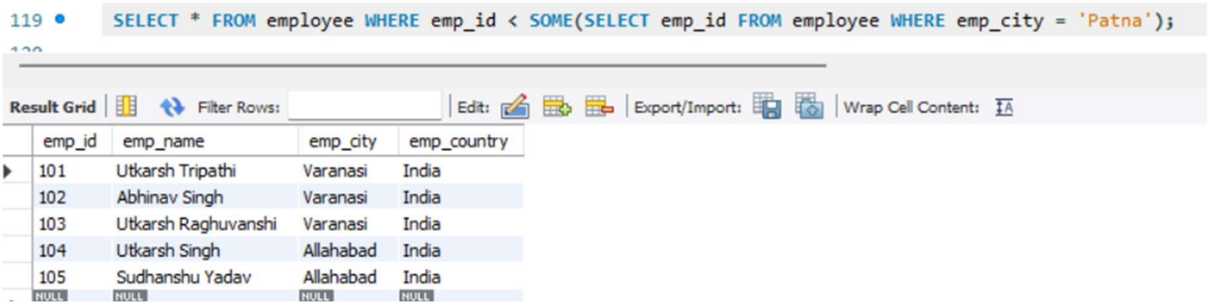
17.EXISTS Command



```sql
SELECT emp_name FROM employee WHERE EXISTS(SELECT emp_id FROM employee WHERE emp_city = 'Patna');
```

emp_name
Utkarsh Tripathi
Abhinav Singh
Utkarsh Raghuvanshi
Utkarsh Singh
Sudhanshu Yadav
Ashutosh Kumar

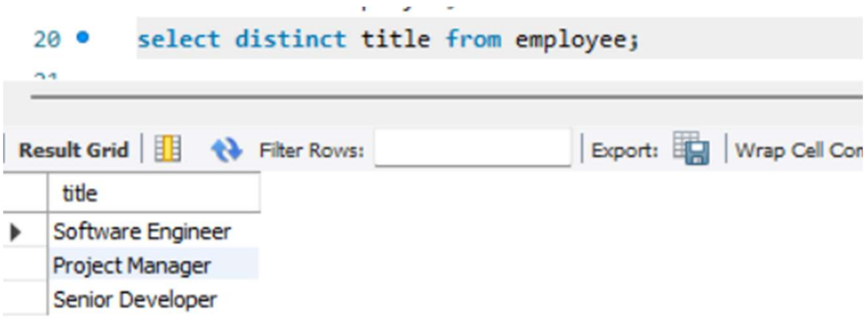
18.SOME Command



```sql
SELECT * FROM employee WHERE emp_id < SOME(SELECT emp_id FROM employee WHERE emp_city = 'Patna');
```

emp_id	emp_name	emp_city	emp_country
101	Utkarsh Tripathi	Varanasi	India
102	Abhinav Singh	Varanasi	India
103	Utkarsh Raghuvanshi	Varanasi	India
104	Utkarsh Singh	Allahabad	India
105	Sudhanshu Yadav	Allahabad	India
NULL	NULL	NULL	NULL

19.DISTINCT Command

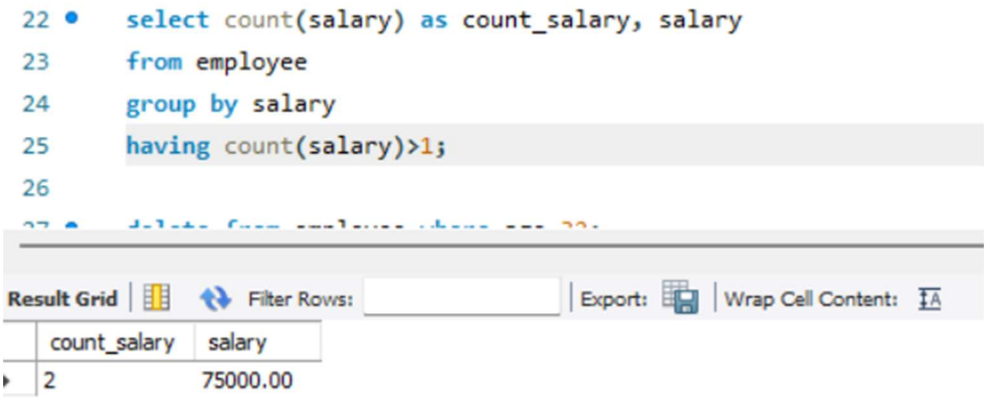


```sql
select distinct title from employee;
```

title
Software Engineer
Project Manager
Senior Developer

20.COUNT Command:

It counts the entity.

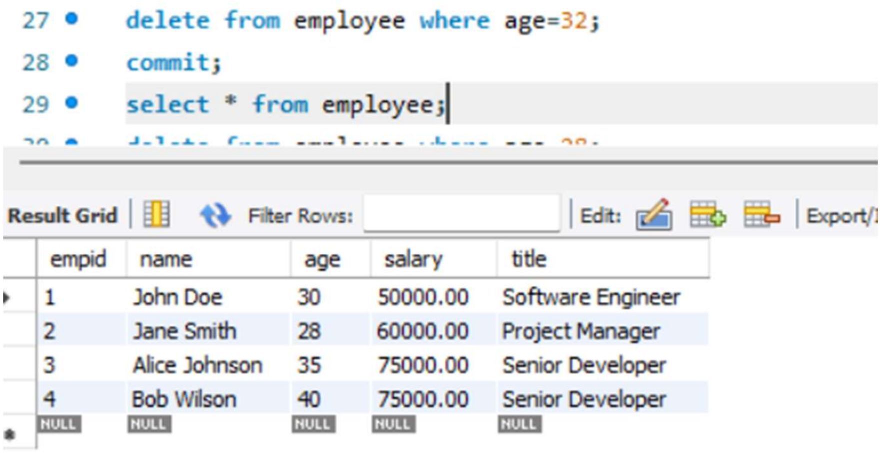


```sql
select count(salary) as count_salary, salary
from employee
group by salary
having count(salary)>1;
```

count_salary	salary
2	75000.00

21.DELETE and COMMIT Command:

The COMMIT command is the transactional command used to save changes invoked by a transaction. It saves all the transactions occurred on the database since the last COMMIT or ROLLBACK

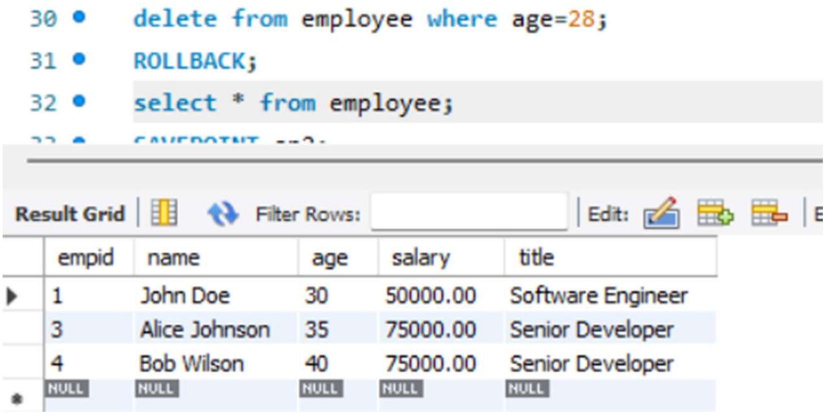


```sql
delete from employee where age=32;
commit;
select * from employee;
```

empid	name	age	salary	title
1	John Doe	30	50000.00	Software Engineer
2	Jane Smith	28	60000.00	Project Manager
3	Alice Johnson	35	75000.00	Senior Developer
4	Bob Wilson	40	75000.00	Senior Developer
NULL	NULL	NULL	NULL	NULL

22.ROLLBACK Command:

The ROLLBACK command is the transactional command used to undo transactions that have not already been saved to the database. This command can only undo transactions since the last COMMIT or ROLLBACK.



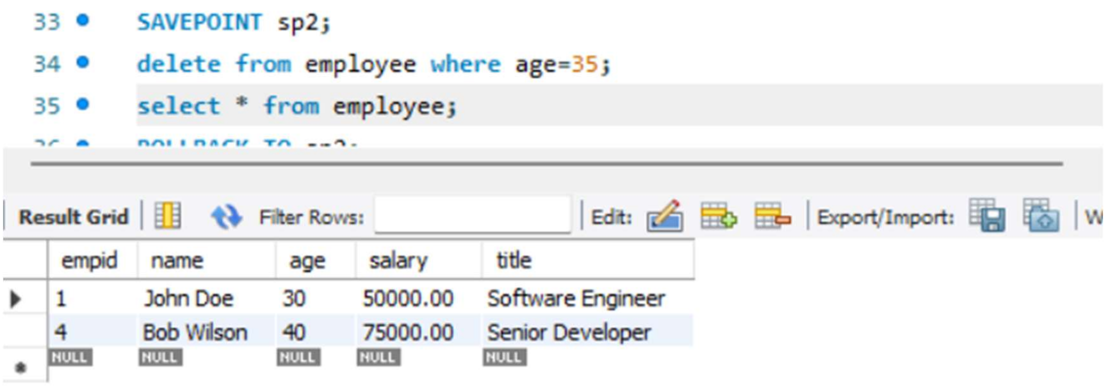
```sql
delete from employee where age=28;
ROLLBACK;
select * from employee;
```

empid	name	age	salary	title
1	John Doe	30	50000.00	Software Engineer
3	Alice Johnson	35	75000.00	Senior Developer
4	Bob Wilson	40	75000.00	Senior Developer
NULL	NULL	NULL	NULL	NULL

23.SAVEPOINT Command:

A SAVEPOINT is a logical rollback point in a transaction.

Usually, when you execute the ROLLBACK command, it undoes the changes until the last COMMIT. But, if you create save points you can partially roll the transaction back to these points. You can create multiple save points between two commits.

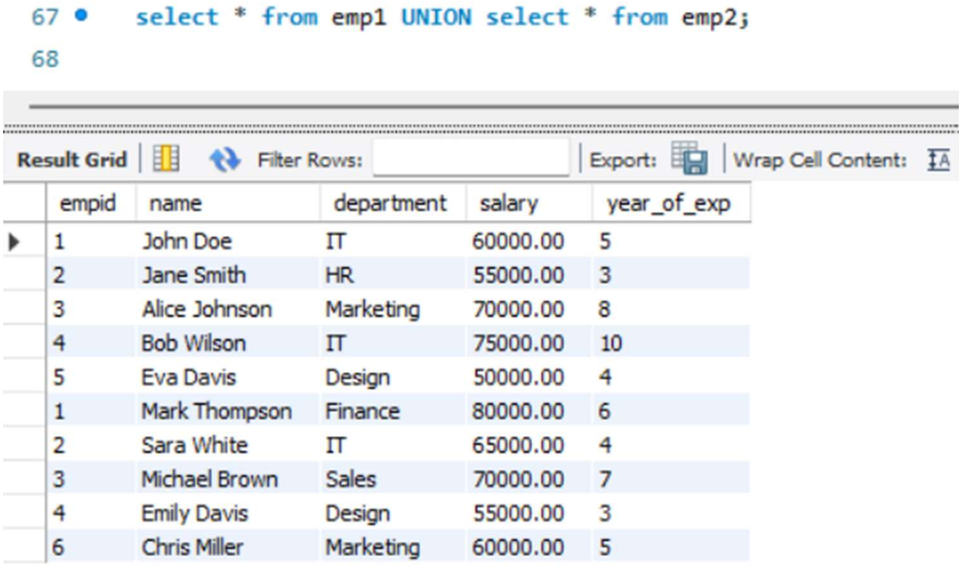


```
SAVEPOINT sp2;
delete from employee where age=35;
select * from employee;
ROLLBACK TO sp2;
```

empid	name	age	salary	title
1	John Doe	30	50000.00	Software Engineer
4	Bob Wilson	40	75000.00	Senior Developer
NULL	NULL	NULL	NULL	NULL

24.UNION Command :

UNION will be used to combine the result of two select statements. Duplicate rows will be eliminated from the results obtained after performing the UNION operation.

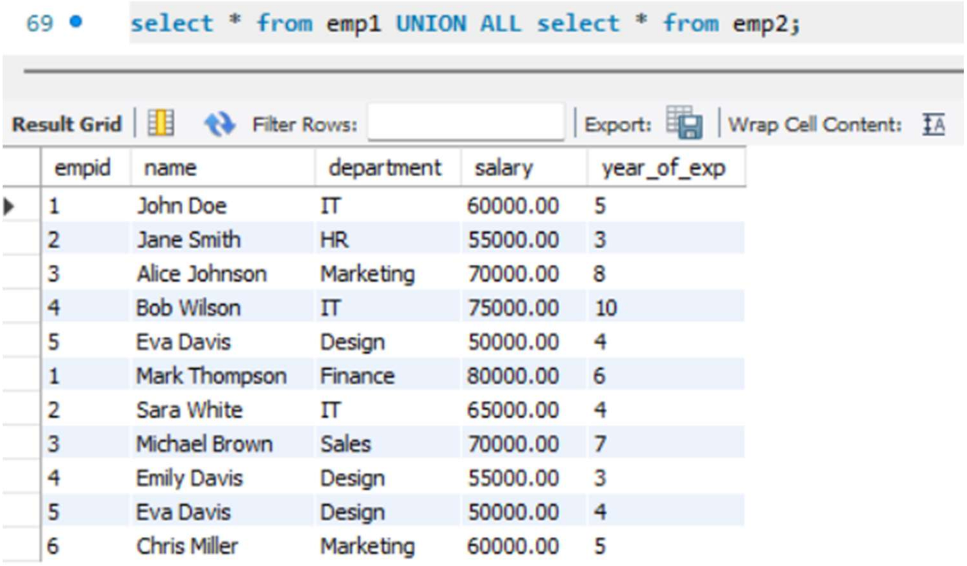


```
select * from emp1 UNION select * from emp2;
```

empid	name	department	salary	year_of_exp
1	John Doe	IT	60000.00	5
2	Jane Smith	HR	55000.00	3
3	Alice Johnson	Marketing	70000.00	8
4	Bob Wilson	IT	75000.00	10
5	Eva Davis	Design	50000.00	4
1	Mark Thompson	Finance	80000.00	6
2	Sara White	IT	65000.00	4
3	Michael Brown	Sales	70000.00	7
4	Emily Davis	Design	55000.00	3
6	Chris Miller	Marketing	60000.00	5

25.UNION ALL Command

Day 3 Assessment



```sql
select * from emp1 UNION ALL select * from emp2;
```

empid	name	department	salary	year_of_exp
1	John Doe	IT	60000.00	5
2	Jane Smith	HR	55000.00	3
3	Alice Johnson	Marketing	70000.00	8
4	Bob Wilson	IT	75000.00	10
5	Eva Davis	Design	50000.00	4
1	Mark Thompson	Finance	80000.00	6
2	Sara White	IT	65000.00	4
3	Michael Brown	Sales	70000.00	7
4	Emily Davis	Design	55000.00	3
5	Eva Davis	Design	50000.00	4
6	Chris Miller	Marketing	60000.00	5