CREATE DATABASE :

Created a database named pet_adoption and to make it active we write use database_name;

```
CREATE DATABASE pet_adoption;
USE pet_adoption;
```

CREATE TABLES

Created two tables animals and adoptions with corresponding attributes and primary keys. We made primary key in animals table to increment automatically.

```
4  CREATE TABLE animals (
5  id integer primary key auto_increment,
6  name varchar(50),
7  breed varchar(50),
8  color varchar(50),
9  gender varchar(50),
10 status INTEGER);
```

```
CREATE TABLE adoptions (
animal_id integer primary key,
name varchar(50),
contact varchar(50),
date TIMESTAMP);
```

SHOW TABLES & COLUMNS

```
18  SHOW TABLES;
```

Result Grid | Filter Rows:

Tables_in_pet_adoption
adoptions
animals

```
19 •  SHOW COLUMNS FROM animals;
```

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

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
name	varchar(50)	YES		NULL	
breed	varchar(50)	YES		NULL	
color	varchar(50)	YES		NULL	
gender	varchar(50)	YES		NULL	
status	int	YES		NULL	

```
20 •  SHOW COLUMNS FROM adoptions;
```

Result Grid | Filter Rows: | Export: | Wra

Field	Type	Null	Key	Default	Extra
animal_id	int	NO	PRI	NULL	
name	varchar(50)	YES		NULL	
contact	varchar(50)	YES		NULL	
date	timestamp	YES		NULL	

INSERT

```
22 •  INSERT INTO animals (name, breed, color, gender, status) VALUES ('Bellyflop', 'Beagle', 'Brown', 'Male', 0);
23 •  INSERT INTO animals (name, breed, color, gender, status) VALUES ('Snowy', 'Husky', 'White', 'Female', 0);
24 •  INSERT INTO animals (name, breed, color, gender, status) VALUES ('Princess', 'Pomeranian', 'Black', 'Female', 0);
25 •  INSERT INTO animals (name, breed, color, gender, status) VALUES ('Cricket', 'Chihuahua', 'Brown', 'Male', 0);
26 •  INSERT INTO animals (name, breed, color, gender, status) VALUES ('Princess', 'Poodle', 'Purple', 'Female', 0);
27 •  INSERT INTO animals (name, breed, color, gender, status) VALUES ('Spot', 'Dalmation', 'Black and White', 'Male', 0);
```

SELECT

Get the full list of all properties of all dogs
Get the breeds of all dogs.
Get the names of only female dogs by including a WHERE clause.
Get the IDs of dogs up for adoption.

```
29 •  SELECT * FROM animals;
```

Result Grid | Filter Rows: | Edit:

id	name	breed	color	gender	status
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
NULL	NULL	NULL	NULL	NULL	NULL

```
29 •  SELECT * FROM animals;
30 •  SELECT breed FROM animals;
```

Result Grid | Filter Rows:

breed
Beagle
Husky
Pomeranian
Chihuahua
Poodle
Dalmation

```
31 •  SELECT name FROM animals WHERE gender = 'Female';
```

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

name
Snowy
Princess
Princess

```
32 •  SELECT id FROM animals WHERE status = 0;
```

Result Grid | Filter Rows: | Edit:

id
1
2
3
4
5
6
NULL

UPDATE & DELETE FROM

BEFORE UPDATE AND DELETE:

```
29 •  SELECT * FROM animals;
```

Result Grid | Filter Rows: | Edit:

id	name	breed	color	gender	status
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
NULL	NULL	NULL	NULL	NULL	NULL

AFTER UPDATE AND DELETE:

Updated the color of animal with id = 2 from white to brown

```
34 •  UPDATE animals SET color = 'Brown' WHERE id = 2;
35 •  SELECT * FROM animals;
```

Result Grid | Filter Rows: | Edit:

id	name	breed	color	gender	status
1	Bellyflop	Beagle	Brown	Male	0
2	Snowy	Husky	Brown	Female	0

Updated the color from purple to brown

```
35 •  UPDATE animals SET color = 'Brown' WHERE color = 'Purple';
36 •  SELECT * FROM animals;
```

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

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

Deleted animal with id=4

```
36 •  DELETE FROM animals WHERE id =4;
37 •  SELECT * FROM animals;
```

Result Grid | Filter Rows: | Edit:

id	name	breed	color	gender	status
1	Bellyflop	Beagle	Brown	Male	0
2	Snowy	Husky	Brown	Female	0
3	Princess	Pomeranian	Black	Female	0
5	Princess	Poodle	Brown	Female	0
6	Spot	Dalmation	Black and White	Male	0
NULL	NULL	NULL	NULL	NULL	NULL

UPDATE & INSERT

```
38 •  UPDATE animals SET status = 1 WHERE id = 6;
39 •  INSERT INTO adoptions (animal_id, name, contact, date) VALUES (6, 'Pinocchio', 'realboy@cockroachlabs.com', NOW());
40 •  SELECT * FROM adoptions;
41
```

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

name	contact	date	animal_id
Pinocchio	realboy@cockroachlabs.com	2024-01-20 22:30:04	6
NULL	NULL	NULL	NULL

```
41 •  UPDATE animals SET status = 1 WHERE id = 2;
42 •  INSERT INTO adoptions (animal_id, name, contact, date) VALUES (2, 'Patalie', 'poodlequeen@cockroachlabs.com', NOW());
43 •  UPDATE animals SET status = 1 WHERE id = 5;
44 •  INSERT INTO adoptions (animal_id, name, contact, date) VALUES (5, 'Ella', 'ellacrew@cockroachlabs.com', NOW());
45 •  SELECT * FROM adoptions;
46
```

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

name	contact	date	animal_id
Patalie	poodlequeen@cockroachlabs.com	2024-01-20 22:32:38	2
Ella	ellacrew@cockroachlabs.com	2024-01-20 22:32:38	5
Pinocchio	realboy@cockroachlabs.com	2024-01-20 22:30:04	6
NULL	NULL	NULL	NULL

ORDER BY

```
47 •  SELECT * FROM adoptions ORDER BY date ASC;
```

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

name	contact	date	animal_id
Pinocchio	realboy@cockroachlabs.com	2024-01-20 22:30:04	6
Patalie	poodlequeen@cockroachlabs.com	2024-01-20 22:32:38	2
Ella	ellacrew@cockroachlabs.com	2024-01-20 22:32:38	5
NULL	NULL	NULL	NULL

ALTER TABLE

```
50 • ALTER TABLE animals ADD COLUMN species VARCHAR(50);
51 • SHOW COLUMNS FROM animals;
```

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

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
name	varchar(50)	YES		NULL	
breed	varchar(50)	YES		NULL	
color	varchar(50)	YES		NULL	
gender	varchar(50)	YES		NULL	
status	int	YES		NULL	
species	varchar(50)	YES		NULL	

SET & UPDATE

```
53 • SET sql_safe_updates = FALSE;
54 • UPDATE animals SET species = 'Dog';
55 • SELECT * FROM animals;
```

Result Grid | Filter Rows: | Edit: | Export/Im

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

Add All Cats to the Database

```
56 • INSERT INTO animals (name, species, breed, color, gender, status) VALUES ('Meowmix', 'Cat', 'Munchkin', 'Yellow', 'Female', 0);
57 • INSERT INTO animals (name, species, breed, color, gender, status) VALUES ('Ash', 'Cat', 'Persian', 'Gray', 'Female', 0);
58 • INSERT INTO animals (name, species, breed, color, gender, status) VALUES ('Tiger', 'Cat', 'Bengal', 'Brown', 'Male', 0);
59 • SELECT * FROM animals;
```

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

id	name	breed	color	gender	status	species
1	Bellyflop	Beagle	Brown	Male	0	Dog
2	Snowy	Husky	Brown	Female	1	Dog
3	Princess	Pomeranian	Black	Female	0	Dog
5	Princess	Poodle	Brown	Female	1	Dog
6	Spot	Dalmation	Black and White	Male	1	Dog
7	Meowmix	Munchkin	Yellow	Female	0	Cat
8	Ash	Persian	Gray	Female	0	Cat
9	Tiger	Bengal	Brown	Male	0	Cat
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Creating a Table for Shelters

```
61 • CREATE TABLE shelters (
62     id INTEGER,
63     name VARCHAR(50),
64     location VARCHAR(50)
65     );
```

Inserting data into shelters table

```
67 • INSERT INTO shelters (id, name, location) VALUES (1, 'Animals 4 Homes', 'Red City');
68 • INSERT INTO shelters (id, name, location) VALUES (2, 'Adopt A Buddy', 'Green Town');
69 • INSERT INTO shelters (id, name, location) VALUES (3, 'Fluffy Animals', 'Blue Hills');
```

```
74 • SELECT * FROM shelters;
```

Result Grid | Filter Rows:

id	name	location
1	Animals 4 Homes	Red City
2	Adopt A Buddy	Green Town
3	Fluffy Animals	Blue Hills

Alter and Update animals table

```
71 • ALTER TABLE animals ADD COLUMN shelter INTEGER;
72 • UPDATE animals SET shelter = 1;
```

Adding more animals

```
76 •  INSERT INTO animals (name, shelter, species, breed, color, gender, status) VALUES ('Snoops', 2, 'Dog', 'Beagle', 'Brown', 'Male', 0);
77 •  INSERT INTO animals (name, shelter, species, breed, color, gender, status) VALUES ('Salt', 2, 'Cat', 'Turkish Angora', 'White', 'Female', 0);
78 •  INSERT INTO animals (name, shelter, species, breed, color, gender, status) VALUES ('Fuzz', 3, 'Dog', 'Papillon', 'Gray', 'Male', 0);
79 •  SELECT * FROM animals;
80
```

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

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
2	Snowy	Husky	Brown	Female	1	Dog	1
3	Princess	Pomeranian	Black	Female	0	Dog	1
5	Princess	Poodle	Brown	Female	1	Dog	1
6	Spot	Dalmation	Black and White	Male	1	Dog	1
7	Meowmix	Munchkin	Yellow	Female	0	Cat	1
8	Ash	Persian	Gray	Female	0	Cat	1
9	Tiger	Bengal	Brown	Male	0	Cat	1
10	Snoops	Beagle	Brown	Male	0	Dog	2
11	Salt	Turkish Angora	White	Female	0	Cat	2
12	Fuzz	Papillon	Gray	Male	0	Dog	3
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

JOINS

```
82 •  SELECT * FROM animals JOIN shelters ON animals.shelter = shelters.id;
83
```

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

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	Brown	Female	1	Dog	1	1	Animals 4 Homes	Red City
3	Princess	Pomeranian	Black	Female	0	Dog	1	1	Animals 4 Homes	Red City
5	Princess	Poodle	Brown	Female	1	Dog	1	1	Animals 4 Homes	Red City
6	Spot	Dalmation	Black and White	Male	1	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 Angora	White	Female	0	Cat	2	2	Adopt A Buddy	Green Town
12	Fuzz	Papillon	Gray	Male	0	Dog	3	3	Fluffy Animals	Blue Hills

getting a list of the adopted animals from the first shelter

```
83 •  SELECT * FROM adoptions JOIN animals ON adoptions.animal_id = animals.id WHERE animals.shelter = 1;
```

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

name	contact	date	animal_id	id	name	breed	color	gender	status	species	shelter
Patalie	poodlequeen@cockroachlabs.com	2024-01-20 22:32:38	2	2	Snowy	Husky	Brown	Female	1	Dog	1
Ella	ellacrew@cockroachlabs.com	2024-01-20 22:32:38	5	5	Princess	Poodle	Brown	Female	1	Dog	1
Pinocchio	realboy@cockroachlabs.com	2024-01-20 22:30:04	6	6	Spot	Dalmation	Black and White	Male	1	Dog	1

INNER JOIN:

```
SELECT ENAME,DNAME FROM [dbo].[EMPLOYEES] AS E
INNER JOIN [dbo].[DEPARTMENT] AS D ON E.DNO = D.DNO;
```

117 %

Results | Messages

	ENAME	DNAME
1	KARTIK	IT
2	SOMYA	SALES
3	RAJNIKANT	IT
4	PRABHAKARAN	IT
5	MANISH	ACCOUNT
6	KARTIK	ACCOUNT
7	RAJEEV	ADMIN
8	SAKTI	MARKETING

LEFT JOIN:

```
SELECT ENAME,DNAME FROM [dbo].[EMPLOYEES] AS E
LEFT JOIN [dbo].[DEPARTMENT] AS D ON E.DNO = D.DNO;
```

117 %

Results | Messages

	ENAME	DNAME
1	KARTIK	IT
2	SOMYA	SALES
3	RAJNIKANT	IT
4	PRABHAKARAN	IT
5	MANISH	ACCOUNT
6	KARTIK	ACCOUNT
7	RAJEEV	ADMIN
8	SAKTI	MARKETING
9	NAMAN	NULL

RIGHT JOIN:

```
SELECT ENAME,DNAME FROM [dbo].[EMPLOYEES] AS E
RIGHT JOIN [dbo].[DEPARTMENT] AS D ON E.DNO = D.DNO;
```

117 %

Results | Messages

	ENAME	DNAME
1	KARTIK	IT
2	RAJNIKANT	IT
3	PRABHAKARAN	IT
4	SOMYA	SALES
5	SAKTI	MARKETING
6	MANISH	ACCOUNT
7	KARTIK	ACCOUNT
8	RAJEEV	ADMIN
9	NULL	RD

FULL JOIN:

```
SELECT ENAME,DNAME FROM [dbo].[EMPLOYEES] AS E
FULL JOIN [dbo].[DEPARTMENT] AS D ON E.DNO = D.DNO;
```

117 %

Results | Messages

	ENAME	DNAME
1	KARTIK	IT
2	SOMYA	SALES
3	RAJNIKANT	IT
4	PRABHAKARAN	IT
5	MANISH	ACCOUNT
6	KARTIK	ACCOUNT
7	RAJEEV	ADMIN
8	SAKTI	MARKETING
9	NAMAN	NULL
10	NULL	RD

Logical Operators in SQL

AND Operator: The AND operator is used to combines two or more conditions but if it is true when all the conditions are satisfied.

```
85 •  SELECT * FROM animals WHERE shelter = 1 AND color = 'Brown';
```

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

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
2	Snowy	Husky	Brown	Female	1	Dog	1
5	Princess	Poodle	Brown	Female	1	Dog	1
9	Tiger	Bengal	Brown	Male	0	Cat	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

IN Operator: It is used to remove the multiple OR conditions in SELECT, INSERT, UPDATE, or DELETE.

```
86 •  SELECT * FROM animals WHERE shelter IN (2,3);
```

Result Grid | Filter Rows: | Edit: | Export/Imp

id	name	breed	color	gender	status	species	shelter
10	Snoops	Beagle	Brown	Male	0	Dog	2
11	Salt	Turkish Angora	White	Female	0	Cat	2
12	Fuzz	Papillon	Gray	Male	0	Dog	3
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

LIKE Operator

In SQL, the LIKE operator is used in the WHERE clause to search for a specified pattern in a column.

- % – It is used for zero or more than one character.
- _ – It is used for only one character means fixed length.

```
87 •  SELECT * FROM animals WHERE name LIKE '%uzz';
```

Result Grid | Filter Rows: | Edit:

id	name	breed	color	gender	status	species	shelter
12	Fuzz	Papillon	Gray	Male	0	Dog	3
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

NOT Operator:

```
88 •  SELECT * FROM animals WHERE name NOT LIKE '%uzz';
```

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

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
2	Snowy	Husky	Brown	Female	1	Dog	1
3	Princess	Pomeranian	Black	Female	0	Dog	1
5	Princess	Poodle	Brown	Female	1	Dog	1
6	Spot	Dalmation	Black and White	Male	1	Dog	1
7	Meowmix	Munchkin	Yellow	Female	0	Cat	1
8	Ash	Persian	Gray	Female	0	Cat	1
9	Tiger	Bengal	Brown	Male	0	Cat	1
10	Snoops	Beagle	Brown	Male	0	Dog	2
11	Salt	Turkish Angora	White	Female	0	Cat	2
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

OR Operator

The OR operator is used to combines two or more conditions but if it is true when one of the conditions are satisfied.

```
89 •  SELECT * FROM animals WHERE breed = 'Beagle' OR gender = 'Male';
```

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

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
6	Spot	Dalmation	Black and White	Male	1	Dog	1
9	Tiger	Bengal	Brown	Male	0	Cat	1
10	Snoops	Beagle	Brown	Male	0	Dog	2
12	Fuzz	Papillon	Gray	Male	0	Dog	3
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

ANY Operator

The ANY operator:

- It returns a boolean value as a result

It returns TRUE if ANY of the subquery values match the condition

```
91 •  SELECT * FROM animals WHERE id = ANY(SELECT id FROM animals WHERE shelter = 1);
```

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

id	name	breed	color	gender	status	species	shelter
1	Bellyflop	Beagle	Brown	Male	0	Dog	1
2	Snowy	Husky	Brown	Female	1	Dog	1
3	Princess	Pomeranian	Black	Female	0	Dog	1
5	Princess	Poodle	Brown	Female	1	Dog	1
6	Spot	Dalmation	Black and White	Male	1	Dog	1
7	Meowmix	Munchkin	Yellow	Female	0	Cat	1
8	Ash	Persian	Gray	Female	0	Cat	1
9	Tiger	Bengal	Brown	Male	0	Cat	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Distinct Clause : We have two animals with same name = princess but distinct clause displayed unique name.

```
92 •  SELECT DISTINCT name FROM animals;
```

Result Grid | Filter Rows: | Expor

name
Bellyflop
Snowy
Princess
Spot
Meowmix
Ash
Tiger
Snoops
Salt
Fuzz

```
95 •  SELECT COUNT(DISTINCT shelter) FROM animals ;
```

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

COUNT(DISTINCT shelter)
3

GROUP BY : The GROUP BY clause is often used with aggregate functions (MAX, SUM, AVG) to group the results by one or more columns

```
96 •  SELECT COUNT(shelter) FROM animals GROUP BY shelter;
```

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

COUNT(shelter)
8
2
1

HAVING CLAUSE : The HAVING clause is used instead of WHERE with aggregate functions.

```
96 •  SELECT COUNT(shelter) FROM animals GROUP BY shelter HAVING COUNT(shelter)>1;
```

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

COUNT(shelter)
8
2