

Part 2: Database setup and SQL

1) Database server

- Specify where you host your database

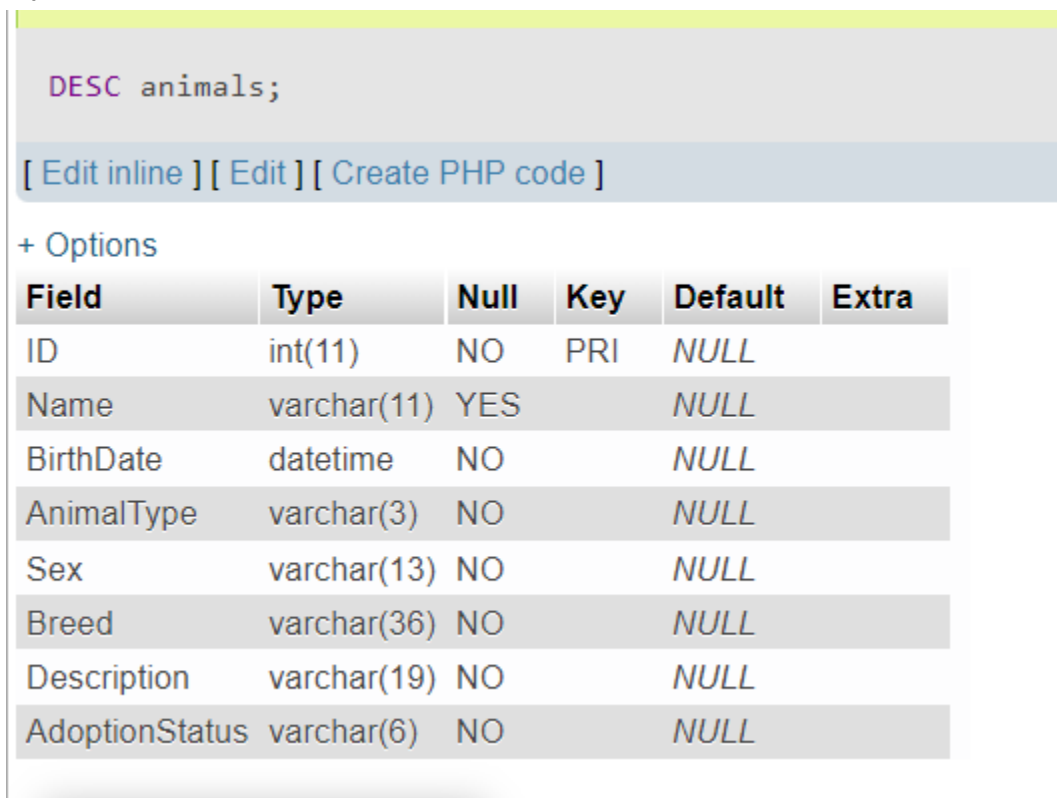
(XAMPP - localhost)

2) Table schemas — For each table,

- Take a screenshot of the table schema and include it in your report.

One way to view a table schema is to run a `DESC table-name` command. The following figure shows an example of a table schema (depending on how you retrieve the table schema, your screen may look different)

- Run a `SELECT COUNT(*) from table-name` to get the total number of rows of data in the table. Include a screenshot showing the number of rows of data in your report.



```
DESC animals;
```

[Edit inline] [Edit] [Create PHP code]

+ Options

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
Name	varchar(11)	YES		NULL	
BirthDate	datetime	NO		NULL	
AnimalType	varchar(3)	NO		NULL	
Sex	varchar(13)	NO		NULL	
Breed	varchar(36)	NO		NULL	
Description	varchar(19)	NO		NULL	
AdoptionStatus	varchar(6)	NO		NULL	

```
SELECT COUNT(*) from animals;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)]

+ Options

COUNT(*)

148

```
DESC dogs;
```

[[Edit inline](#)] [[Edit](#)] [[Create PHP code](#)]

+ Options

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
DogFriendly	varchar(3)	NO		NULL	
ChildFriendly	varchar(3)	NO		NULL	
Size	varchar(6)	NO		NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from dogs;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)]

+ Options

COUNT(*)

76

```
DESC cats;
```

[[Edit inline](#)] [[Edit](#)] [[Create PHP code](#)]

+ Options

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
IndoorOnly	varchar(3)	NO		NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from cats;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)]

+ Options

COUNT(*)

55

```
DESC accounts;
```

[[Edit inline](#)] [[Edit](#)] [[Create PHP code](#)]

+ Options

Field	Type	Null	Key	Default	Extra
phone	bigint(20)	NO	PRI	NULL	
first_name	varchar(12)	NO		NULL	
last_name	varchar(12)	NO		NULL	
photo_id	int(11)	NO		NULL	
hashed_pass	varchar(8)	NO		NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from accounts;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create](#)

+ Options

COUNT(*)

80

```
DESC employees;
```

[[Edit inline](#)] [[Edit](#)] [[Create PHP code](#)]

+ Options

Field	Type	Null	Key	Default	Extra
employee_id	int(11)	NO	PRI	NULL	
phone	int(11)	NO		NULL	
age	int(11)	NO		NULL	
salary	int(11)	NO		NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from employees;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create](#)

+ Options

COUNT(*)

30

```
desc users;
```

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

+ Options

Field	Type	Null	Key	Default	Extra
phone	bigint(20)	NO	PRI	NULL	
address	varchar(40)	NO		NULL	
city	varchar(17)	NO		NULL	
county	varchar(20)	NO		NULL	
state	varchar(2)	NO		NULL	
zip	int(11)	NO		NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from users;
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#)

+ Options

COUNT(*)

50

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

```
desc oversee;
```

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

+ Options

Field	Type	Null	Key	Default	Extra
employee_id	int(11)	NO		NULL	
animal_id	int(11)	NO	PRI	NULL	
start_date	date	NO		NULL	
end_date	date	NO		NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from oversee;
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#)

+ Options

COUNT(*)

148

```
desc animal_photo;
```

[\[Edit inline \]](#) [\[Edit \]](#) [\[Create PHP code \]](#)

+ Options

Field	Type	Null	Key	Default	Extra
animals_id	int(11)	NO	PRI	NULL	
photo_id	int(11)	NO	PRI	NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from animal_photo;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)]

+ Options

COUNT(*)

148

```
desc adopted;
```

[[Edit inline](#)] [[Edit](#)] [[Create PHP code](#)]

+ Options

Field	Type	Null	Key	Default	Extra
animal_id	int(11)	NO	PRI	NULL	
phone	varchar(14)	YES		NULL	
date_of_adoption	date	YES		NULL	
AdoptionStatus	varchar(6)	NO		NULL	

Your SQL query has been executed successfully.

```
SELECT COUNT(*) from adopted;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)]

+ Options

COUNT(*)

148

3) Describe, justify, and provide evidence showing that your data are **practical and sufficient**.

- Include sample data of each table.

(sample data on github)

- Although you are not required to submit the entire dataset, the number of rows (records) of data in each table must be adequate.
- Showing a few rows of data is typically considered lack of evidence and thus insufficient.

Accounts:

There are a decent number of accounts since it is the superclass of both users and employees.

Showing rows 0 - 79 (80 total, Query took 0.0011 seconds.)

SELECT * FROM `accounts`






































































☐ Profiling
[Edit inline]
[Edit]
[Explain SQL]
[Create PHP code]
[Refresh]

☐ Show all
Number of rows:
250
Filter rows:
Search this table
Sort by key:
N

+ Options

				phone	first_name	last_name	photo_id	hashed_pass
<input type="checkbox"/>	Edit	Copy	Delete	2017096245	Ernie	Stenseth	1032	2D85C1B6
<input type="checkbox"/>	Edit	Copy	Delete	2088625339	Vallie	Mondella	1069	AF4A996
<input type="checkbox"/>	Edit	Copy	Delete	2108129597	Valentine	Gillian	1038	3926F6F9
<input type="checkbox"/>	Edit	Copy	Delete	2124029216	Brock	Bologna	1050	6961020
<input type="checkbox"/>	Edit	Copy	Delete	2125824976	Willow	Kusko	1028	F885CCD
<input type="checkbox"/>	Edit	Copy	Delete	2126749610	Tawna	Buven	1065	196A54F5
<input type="checkbox"/>	Edit	Copy	Delete	2126758570	Jose	Stockham	1036	315A5E70
<input type="checkbox"/>	Edit	Copy	Delete	2128601579	Alishia	Sergi	1034	3422187A
<input type="checkbox"/>	Edit	Copy	Delete	2152119589	Deeanna	Juhas	1059	DFAA437
<input type="checkbox"/>	Edit	Copy	Delete	2158741229	Minna	Amigon	1010	21405D19
<input type="checkbox"/>	Edit	Copy	Delete	2159079111	Blair	Malet	1049	73A8545
<input type="checkbox"/>	Edit	Copy	Delete	2159348655	Amber	Monarrez	1056	59C34D0
<input type="checkbox"/>	Edit	Copy	Delete	3037247371	Carmelina	Lindall	1063	C833714
<input type="checkbox"/>	Edit	Copy	Delete	3056067291	Lavera	Perin	1043	3AD48CBF
<input type="checkbox"/>	Edit	Copy	Delete	3104985651	Kiley	Caldarera	1012	854B6AB
<input type="checkbox"/>	Edit	Copy	Delete	3132887937	Yuki	Whobrey	1018	20DF55B1

Console

<input type="checkbox"/>				3234532780	Kanisha	Waycott	1047	2706C528
<input type="checkbox"/>				4014582547	Delmy	Ahle	1058	7DB7579
<input type="checkbox"/>				4019608259	Blondell	Pugh	1060	4A727FC
<input type="checkbox"/>				4085401785	Veronika	Inouye	1021	17FA0025
<input type="checkbox"/>				4087523500	Leota	Dilliard	1007	33460528
<input type="checkbox"/>				4105204832	Laurel	Reitler	1076	294BAAD7
<input type="checkbox"/>				4106558723	Kris	Marrier	1009	21531BD7
<input type="checkbox"/>				4106691642	Ezekiel	Chui	1027	2A8A1497
<input type="checkbox"/>				4109149018	Ilene	Eroman	1068	16962391
<input type="checkbox"/>				4146619598	Gladys	Rim	1017	37C53E47
<input type="checkbox"/>				4147481374	Maurine	Yglesias	1064	14E374C7
<input type="checkbox"/>				4153152761	Kallie	Blackwood	1070	22475E70
<input type="checkbox"/>				4195032484	Simona	Morasca	1005	1B0CCFEF
<input type="checkbox"/>				4407808425	Graciela	Ruta	1013	24714171
<input type="checkbox"/>				5046218927	James	Butt	1000	20F7F4CD
<input type="checkbox"/>				5049799175	Solange	Shinko	1035	2DE1937C
<input type="checkbox"/>				5059773911	Francine	Vocelka	1031	DECEDF8
<input type="checkbox"/>				5124863817	Cecily	Hollack	1062	592CB9D
<input type="checkbox"/>				5135701893	Donette	Foller	1004	1A5C3162
<input type="checkbox"/>				5169686051	Tonette	Wenner	1055	2F0CA9FF
<input type="checkbox"/>				5189667987	Maryann	Royster	1023	1C8DE146
<input type="checkbox"/>				5415488197	Youlanda	Schemmer	1040	36A10D25
<input type="checkbox"/>				5858668313	Moon	Parlato	1075	282965E5

Adopted:

There are the same amount of animals for adoption since every animal is either adopted, fostered, or not adopted. In general, there are a lot of animals since the database keeps track of animals still in the shelter and who have found a loving home.

















































✓ Showing rows 0 - 147 (148 total, Query took 0.0008 seconds.)






































































SELECT * FROM `adopted`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: Filter rows: Sort by key

+ Options

<div><div><div></div><div></div><div></div></div></div>				animal_id	phone	date_of_adoption	AdoptionStatus	
<input type="checkbox"/>		Edit	 Copy	 Delete	1	2017096245	2021-01-07	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	2	2088625339	2021-02-25	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	3	2108129597	2021-03-14	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	4	2124029216	2021-04-26	Adopt
<input type="checkbox"/>		Edit	 Copy	 Delete	5	2126749610	2021-07-17	Adopt
<input type="checkbox"/>		Edit	 Copy	 Delete	6	2126758570	2021-07-19	Adopt
<input type="checkbox"/>		Edit	 Copy	 Delete	7	2128601579	2021-04-28	Adopt
<input type="checkbox"/>		Edit	 Copy	 Delete	8	2152119589	2021-04-08	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	9	2159079111	2021-05-18	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	10	2159348655	2021-03-05	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	11	3037247371	2021-06-08	Adopt
<input type="checkbox"/>		Edit	 Copy	 Delete	12	3056067291	2021-04-28	Adopt
<input type="checkbox"/>		Edit	 Copy	 Delete	13	3234532780	2021-07-15	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	14	4014582547	2021-06-21	Foster
<input type="checkbox"/>		Edit	 Copy	 Delete	15	4019608259	2021-10-02	Foster
<input checked="" type="checkbox"/>	 Console	it	 Copy	 Delete	16	4105204832	2021-03-05	Adopt

<div><div>← T →</div><div>▼</div></div>				animal_id	phone	date_of_adoption	AdoptionStatus
<input type="checkbox"/>	 Edit	 Copy	 Delete	17	4109149018	2021-07-06	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	18	4147481374	2021-06-15	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	19	4153152761	2021-08-21	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	20	5049799175	2021-10-06	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	21	5059773911	2021-07-04	Foster
<input type="checkbox"/>	 Edit	 Copy	 Delete	22	5124863817	2021-06-16	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	23	5169686051	2021-01-08	Foster
<input type="checkbox"/>	 Edit	 Copy	 Delete	24	5415488197	2021-06-08	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	25	5858668313	2021-04-26	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	26	6083367444	2021-03-27	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	27	6108145533	2021-08-03	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	28	6148019788	2021-03-27	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	29	6173995124	2021-02-05	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	30	6505285783	2021-04-28	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	31	7322341546	2021-06-30	Foster
<input type="checkbox"/>	 Edit	 Copy	 Delete	32	7329247882	2021-03-27	Foster
<input type="checkbox"/>	 Edit	 Copy	 Delete	33	7403438575	2021-09-14	Foster
<input type="checkbox"/>	 Edit	 Copy	 Delete	34	7734465569	2021-09-09	Foster
<input type="checkbox"/>	 Edit	 Copy	 Delete	35	7854637829	2021-01-07	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	36	8039255213	2021-03-03	Foster
<input type="checkbox"/>	 Edit	 Copy	 Delete	37	8058326163	2021-06-02	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	38	8143935571	2021-03-04	Adopt
<input type="checkbox"/>	 Edit	 Copy	 Delete	39	8184234007	2021-05-20	Adopt

Console

Console

Animals:

There are many animals since every row is every animal who has ever been in the shelter.

✓ Showing rows 0 - 99 (148 total, Query took 0.0008 seconds.)

```
SELECT * FROM `animal_photo`
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)]

1 ▾

>

>>

☐ Show all

Number of rows:

1

+ Options








































































animals_id

photo_id

<input type="checkbox"/>	Edit	Copy	Delete	1	1
<input type="checkbox"/>	Edit	Copy	Delete	2	2
<input type="checkbox"/>	Edit	Copy	Delete	3	3
<input type="checkbox"/>	Edit	Copy	Delete	4	4
<input type="checkbox"/>	Edit	Copy	Delete	5	5
<input type="checkbox"/>	Edit	Copy	Delete	6	6
<input type="checkbox"/>	Edit	Copy	Delete	7	7
<input type="checkbox"/>	Edit	Copy	Delete	8	8
<input type="checkbox"/>	Edit	Copy	Delete	9	9
<input type="checkbox"/>	Edit	Copy	Delete	10	10
<input type="checkbox"/>	Edit	Copy	Delete	11	11
<input type="checkbox"/>	Edit	Copy	Delete	12	12
<input type="checkbox"/>	Edit	Copy	Delete	13	13
<input type="checkbox"/>	Edit	Copy	Delete	14	14
<input type="checkbox"/>	Edit	Copy	Delete	15	15
<input type="checkbox"/>	Edit	Copy	Delete	16	16

Console

<input type="checkbox"/>	 Edit	 Copy	 Delete	16	16
<input type="checkbox"/>	 Edit	 Copy	 Delete	17	17
<input type="checkbox"/>	 Edit	 Copy	 Delete	18	18
<input type="checkbox"/>	 Edit	 Copy	 Delete	19	19
<input type="checkbox"/>	 Edit	 Copy	 Delete	20	20
<input type="checkbox"/>	 Edit	 Copy	 Delete	21	21
<input type="checkbox"/>	 Edit	 Copy	 Delete	22	22
<input type="checkbox"/>	 Edit	 Copy	 Delete	23	23
<input type="checkbox"/>	 Edit	 Copy	 <u>Delete</u>	24	24
<input type="checkbox"/>	 Edit	 Copy	 Delete	25	25
<input type="checkbox"/>	 Edit	 Copy	 Delete	26	26
<input type="checkbox"/>	 Edit	 Copy	 Delete	27	27
<input type="checkbox"/>	 Edit	 Copy	 Delete	28	28
<input type="checkbox"/>	 Edit	 Copy	 Delete	29	29
<input type="checkbox"/>	 Edit	 Copy	 Delete	30	30
<input type="checkbox"/>	 Edit	 Copy	 Delete	31	31
<input type="checkbox"/>	 Edit	 Copy	 Delete	32	32
<input type="checkbox"/>	 Edit	 Copy	 Delete	33	33
<input type="checkbox"/>	 Edit	 Copy	 Delete	34	34
<input type="checkbox"/>	 Edit	 Copy	 Delete	35	35
<input type="checkbox"/>	 Edit	 Copy	 Delete	36	36
<input type="checkbox"/>	 Edit	 Copy	 Delete	37	37
<input type="checkbox"/>	 Edit	 Copy	 Delete	38	38

Cats:

This table keeps track of all of the cats in animals.

✓ Showing rows 0 - 24 (55 total, Query took 0.0005 seconds.)

```
SELECT * FROM `cats`
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)]

1 ▾

>

>>

☐

Show all














Number of rows:

+ Options

					ID	IndoorOnly
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	7	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	8	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	19	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	20	Yes	
<input type="checkbox"/>	 Edit	 Copy	 Delete	21	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	22	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	24	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	26	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	27	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	33	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	34	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	35	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	37	Yes	
<input type="checkbox"/>	 Edit	 Copy	 Delete	42	No	
<input type="checkbox"/>	 Edit	 Copy	 Delete	43	No	



Console

<input type="checkbox"/>	 Edit	 Copy	 Delete	26	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	27	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	33	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	34	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	35	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	37	Yes
<input type="checkbox"/>	 Edit	 Copy	 Delete	42	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	43	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	44	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	45	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	47	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	51	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	53	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	54	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	56	Yes
<input type="checkbox"/>	 Edit	 Copy	 Delete	58	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	59	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	63	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	64	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	65	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	67	Yes
<input type="checkbox"/>	 Edit	 Copy	 Delete	70	No
<input type="checkbox"/>	 Edit	 Copy	 Delete	73	No

Dogs:

This table keeps track of all the dogs in animals.

✓ Showing rows 0 - 24 (76 total, Query took 0.0006 seconds.)

SELECT * FROM `dogs`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

1 ▾

>

>>

☐


















































Show all

Number of rows:




25 ▾

Filter rows

+ Options

				▼	ID	DogFriendly	ChildFriendly	Size
<input type="checkbox"/>	 Edit	 Copy	 Delete		1	Yes	Yes	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete		2	No	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete		4	No	No	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete		5	Yes	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete		6	Yes	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete		9	Yes	No	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete		10	Yes	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete		11	No	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete		12	No	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete		13	Yes	Yes	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete		14	No	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete		15	No	No	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete		16	Yes	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete		17	No	Yes	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete		18	Yes	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete		23	Yes	No	Medium

 Console

<input type="checkbox"/>	 Edit	 Copy	 Delete	23	Yes	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	25	Yes	No	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete	28	Yes	Yes	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	29	Yes	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	30	Yes	Yes	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	31	No	No	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete	32	Yes	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	36	Yes	Yes	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	38	Yes	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	39	No	No	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete	40	Yes	Yes	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	41	No	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	46	No	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	48	No	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	49	No	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	50	No	Yes	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	52	No	No	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	57	Yes	Yes	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete	60	Yes	Yes	Medium
<input type="checkbox"/>	 Edit	 Copy	 Delete	61	No	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	62	No	No	Small
<input type="checkbox"/>	 Edit	 Copy	 Delete	66	Yes	No	Large
<input type="checkbox"/>	 Edit	 Copy	 Delete	68	No	No	Medium

Employees:

This table lists all of the employees from accounts.

✓ Showing rows 0 - 24 (30 total, Query took 0.0012 seconds.)

```
SELECT * FROM `employees`
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

1 ▾

>

>>

☐ Show all

Number of rows:

25 ▾

Filter

+ Options



employee_id

phone

age

salary



Edit



Copy



Delete

0

2147483647

39

44000



Edit



Copy



Delete

1

2147483647

34

39000



Edit



Copy



Delete

2

2147483647

36

41000



Edit



Copy



Delete

3

2147483647

23

28000



Edit



Copy



Delete

4

2147483647

28

33000



Edit



Copy



Delete

5

2147483647

24

29000



Edit



Copy



Delete

6

2147483647

33

38000



Edit



Copy



Delete

7

2147483647

25

30000



Edit



Copy



Delete

8

2147483647

25

30000



Edit



Copy



Delete

9

2147483647

30

35000



Edit



Copy



Delete

10

2147483647

38

43000



Edit



Copy



Delete

11




































2147483647

33

38000



Console

<input type="checkbox"/>	 Edit	 Copy	 Delete	12	2147483647	22	27000
<input type="checkbox"/>	 Edit	 Copy	 Delete	13	2147483647	28	33000
<input type="checkbox"/>	 Edit	 Copy	 Delete	14	2147483647	25	30000
<input type="checkbox"/>	 Edit	 Copy	 Delete	15	2147483647	31	36000
<input type="checkbox"/>	 Edit	 Copy	 Delete	16	2147483647	30	35000
<input type="checkbox"/>	 Edit	 Copy	 Delete	17	2147483647	30	35000
<input type="checkbox"/>	 Edit	 Copy	 Delete	18	2147483647	26	31000
<input type="checkbox"/>	 Edit	 Copy	 Delete	19	2147483647	38	43000
<input type="checkbox"/>	 Edit	 Copy	 Delete	20	2147483647	28	33000
<input type="checkbox"/>	 Edit	 Copy	 Delete	21	2147483647	37	42000
<input type="checkbox"/>	 Edit	 Copy	 Delete	22	2147483647	21	26000
<input type="checkbox"/>	 Edit	 Copy	 Delete	23	2147483647	26	31000
<input type="checkbox"/>	 Edit	 Copy	 Delete	24	2147483647	26	31000

Oversee:

This table shows all of the relationships between shelter animal and employee.

✓ Showing rows 0 - 147 (148 total, Query took 0.0009 seconds.)

SELECT * FROM `oversee`



























☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☒ Show all | Number of rows: All Filter rows:

+ Options

<div><div><div><div></div><div></div><div></div></div></div><div></div></div>				employee_id	animal_id	start_date	end_date
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	0	1	2021-12-13	2021-12-27			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	0	2	2022-04-17	2022-05-01			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	0	3	2022-05-06	2022-05-20			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	0	4	2021-11-30	2021-12-14			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	0	5	2021-11-30	2021-12-14			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	1	6	2022-02-13	2022-02-27			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	1	7	2021-11-25	2021-12-09			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	1	8	2022-05-07	2022-05-21			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	1	9	2022-02-13	2022-02-27			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	1	10	2022-05-14	2022-05-28			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	2	11	2022-02-22	2022-03-08			
<div><div><div></div></div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>	2	12	2022-01-29	2022-02-12			

 Console

				employee_id	animal_id	start_date	end_date
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	13	2022-02-16	2022-03-02
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	14	2022-01-25	2022-02-08
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	15	2022-05-14	2022-05-28
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	16	2021-11-17	2021-12-01
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	17	2022-02-16	2022-03-02
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	18	2021-12-26	2022-01-09
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	19	2022-04-25	2022-05-09
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	20	2021-12-04	2021-12-18
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	21	2022-03-02	2022-03-16
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	22	2021-11-11	2021-11-25
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	23	2022-01-13	2022-01-27
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	24	2021-12-04	2021-12-18
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	25	2022-01-17	2022-01-31
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	26	2022-03-25	2022-04-08
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	27	2021-12-04	2021-12-18
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	28	2021-12-12	2021-12-26
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	29	2021-12-30	2022-01-13
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	30	2022-05-08	2022-05-22
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	31	2021-12-26	2022-01-09

Users:

This table shows all of the users who are interacting with a shelter animal.

✓ Showing rows 0 - 49 (50 total, Query took 0.0005 seconds.)

`SELECT * FROM `users``

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☒ Show all | Number of rows: All | Filter rows: | Sort by key: None

+ Options

			phone	address	city	county	state	zip
<input type="checkbox"/>				2017096245	45 E Liberty St	Ridgefield Park	Bergen	NJ 7660
<input type="checkbox"/>				2088625339	74 W College St	Boise	Ada	ID 83707
<input type="checkbox"/>				2108129597	775 W 17th St	San Antonio	Bexar	TX 78204
<input type="checkbox"/>				2124029216	4486 W O St #1	New York	New York	NY 10003
<input type="checkbox"/>				2126749610	3305 Nabell Ave #679	New York	New York	NY 10009
<input type="checkbox"/>				2126758570	128 Bransten Rd	New York	New York	NY 10011
<input type="checkbox"/>				2128601579	2742 Distribution Way	New York	New York	NY 10025
<input type="checkbox"/>				2152119589	14302 Pennsylvania Ave	Huntingdon Valley	Montgomery	PA 19006
<input type="checkbox"/>				2159079111	209 Decker Dr	Philadelphia	Philadelphia	PA 19132
<input type="checkbox"/>				2159348655	14288 Foster Ave #4121	Jenkintown	Montgomery	PA 19046
<input type="checkbox"/>				3037247371	2664 Lewis Rd	Littleton	Douglas	CO 80126
				3056067291	678 3rd Ave	Miami	Miami-Dade	FL 33196
<input type="checkbox"/>				3234532780	5 Tomahawk Dr	Los Angeles	Los Angeles	CA 90006
<input type="checkbox"/>				4014582547	65895 S 16th St	Providence	Providence	RI 2909
<input type="checkbox"/>				4019608259	201 Hawk Ct	Providence	Providence	RI 2904
<input type="checkbox"/>				4105204832	6 Kains Ave	Baltimore	Baltimore City	MD 21215
<input type="checkbox"/>				4109149018	2853 S Central Expy	Glen Burnie	Anne Arundel	MD 21061
<input type="checkbox"/>				4147481374	59 Shady Ln #53	Milwaukee	Milwaukee	WI 53214
<input type="checkbox"/>				4153152761	701 S Harrison Rd	San Francisco	San Francisco	CA 94104
<input type="checkbox"/>				5049799175	426 Wolf St	Metairie	Jefferson	LA 70002
<input type="checkbox"/>				5059773911	366 South Dr	Las Cruces	Dona Ana	NM 88011
<input type="checkbox"/>				5124863817	59 N Groesbeck Hwy	Austin	Travis	TX 78731
<input type="checkbox"/>				5169686051	4545 Courthouse Rd	Westbury	Nassau	NY 11590
<input type="checkbox"/>				5415488197	2881 Lewis Rd	Prineville	Crook	OR 97754
<input type="checkbox"/>				5858668313	74989 Brandon St	Wellsville	Allegany	NY 14895
<input type="checkbox"/>				6083367444	762 S Main St	Madison	Dane	WI 53711
<input type="checkbox"/>				6108145533	71 San Mateo Ave	Wayne	Delaware	PA 19087
<input type="checkbox"/>				6148019788	74874 Atlantic Ave	Columbus	Franklin	OH 43215
<input type="checkbox"/>				6173995124	38938 Park Blvd	Boston	Suffolk	MA 2128
<input type="checkbox"/>				6505285783	30 W 80th St #1995	San Carlos	San Mateo	CA 94070
				7322341546	53075 Sw 152nd Ter #615	Monroe Township	Middlesex	NJ 8831

4)List of all SQL commands used in your app

- Non-advanced SQL commands. Please refer to the [Project proposal and UI design](#) page for details and minimum requirements. These include commands to
 - Create tables
 - Retrieve data
 - Add data
 - Update data
 - Delete data
- Advanced SQL commands

CREATE TABLES & ADD DATA:

- **accounts.sql**

```
CREATE TABLE Accounts(
  phone      BIGINT NOT NULL PRIMARY KEY
  ,first_name VARCHAR(12) NOT NULL
  ,last_name  VARCHAR(12) NOT NULL
  ,photo_id   INTEGER NOT NULL
  ,hashed_pass VARCHAR(8) NOT NULL
);
```

```
INSERT INTO Accounts(phone,first_name,last_name,photo_id,hashed_pass)
VALUES (5046218927,'James','Butt',1000,'20F7F4CD');
INSERT INTO Accounts(phone,first_name,last_name,photo_id,hashed_pass)
VALUES (8102929388,'Josephine','Darakjy',1001,'21BB7792');
INSERT INTO Accounts(phone,first_name,last_name,photo_id,hashed_pass)
VALUES (8566368749,'Art','Venere',1002,'8606B31');
```

```
.
. check accounts.sql for all add data commands into the Accounts table
.
```

```
INSERT INTO Accounts(phone,first_name,last_name,photo_id,hashed_pass)
VALUES (9733542040,'Delisa','Crupi',1077,'3A3B890C');
INSERT INTO Accounts(phone,first_name,last_name,photo_id,hashed_pass)
VALUES (7734465569,'Viva','Toelkes',1078,'38CB55D5');
INSERT INTO Accounts(phone,first_name,last_name,photo_id,hashed_pass)
VALUES (9739273447,'Elza','Lipke',1079,'221221A7');
```

- **animals.sql**

```
CREATE TABLE Animals(
  ID          INTEGER NOT NULL PRIMARY KEY
```



```
,Name      VARCHAR(11)
,BirthDate  DATE NOT NULL
,AnimalType VARCHAR(3) NOT NULL
,Sex        VARCHAR(13) NOT NULL
,Breed      VARCHAR(36) NOT NULL
,Description VARCHAR(19) NOT NULL
,AdoptionStatus VARCHAR(6) NOT NULL
);
```

```
INSERT INTO
Animals(ID,Name,BirthDate,AnimalType,Sex,Breed,Description,AdoptionStatus)
VALUES (1,'Summer','2015-10-12','Dog','Intact Female','Labrador Retriever
Mix','Red/White','Foster');
```

```
INSERT INTO
Animals(ID,Name,BirthDate,AnimalType,Sex,Breed,Description,AdoptionStatus)
VALUES (2,'Cheyenne','2014-07-26','Dog','Spayed Female','German
Shepherd/Siberian
Husky','Black/Tan','Foster');
```

```
INSERT INTO
Animals(ID,Name,BirthDate,AnimalType,Sex,Breed,Description,AdoptionStatus)
VALUES (3,'Gus','2016-01-13','Cat','Neutered Male','Domestic Shorthair
Mix','Brown Tabby','Foster');
```

```
.
. check animals.sql for all add data commands into the Animals table
```

```
INSERT INTO
Animals(ID,Name,BirthDate,AnimalType,Sex,Breed,Description,AdoptionStatus)
VALUES (146,'Molly','2015-10-27','Dog','Spayed Female','Pit Bull Mix','Black
Brindle/White','Foster');
```

```
INSERT INTO
Animals(ID,Name,BirthDate,AnimalType,Sex,Breed,Description,AdoptionStatus)
VALUES (147,'Toby','2014-08-12','Cat','Neutered Male','Snowshoe Mix','Lilac
Point','Foster');
```

```
INSERT INTO
Animals(ID,Name,BirthDate,AnimalType,Sex,Breed,Description,AdoptionStatus)
VALUES (148,'Maximillion','2014-11-12','Dog','Neutered Male','Miniature Pinscher
Mix','Black/Brown','Adopt');
```

- **cats.sql**

```
CREATE TABLE Cats(  
  ID          INTEGER NOT NULL PRIMARY KEY  
  ,IndoorOnly VARCHAR(3) NOT NULL  
);
```

```
INSERT INTO Cats(ID,IndoorOnly) VALUES (3,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (7,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (8,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (19,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (20,'Yes');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (21,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (22,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (24,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (26,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (27,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (33,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (34,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (35,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (37,'Yes');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (42,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (43,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (44,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (45,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (47,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (51,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (53,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (54,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (56,'Yes');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (58,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (59,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (63,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (64,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (65,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (67,'Yes');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (70,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (73,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (75,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (76,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (77,'No');  
INSERT INTO Cats(ID,IndoorOnly) VALUES (78,'No');
```

```

INSERT INTO Cats(ID,IndoorOnly) VALUES (80,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (90,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (93,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (99,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (101,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (102,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (105,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (109,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (110,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (121,'Yes');
INSERT INTO Cats(ID,IndoorOnly) VALUES (122,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (123,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (128,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (129,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (130,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (132,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (135,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (137,'Yes');
INSERT INTO Cats(ID,IndoorOnly) VALUES (142,'No');
INSERT INTO Cats(ID,IndoorOnly) VALUES (147,'Yes');

```

- **dogs.sql**

```

CREATE TABLE Dogs(
    ID          INTEGER NOT NULL PRIMARY KEY
    ,DogFriendly VARCHAR(3) NOT NULL
    ,ChildFriendly VARCHAR(3) NOT NULL
    ,Size        VARCHAR(6) NOT NULL
);

```

```

INSERT INTO Dogs(ID,DogFriendly,ChildFriendly,Size) VALUES
(1,'Yes','Yes','Medium');
INSERT INTO Dogs(ID,DogFriendly,ChildFriendly,Size) VALUES
(2,'No','No','Medium');
INSERT INTO Dogs(ID,DogFriendly,ChildFriendly,Size) VALUES
(4,'No','No','Small');

```

.

. *check dogs.sql for all add data commands into the Dogs table*

.

```

INSERT INTO Dogs(ID,DogFriendly,ChildFriendly,Size) VALUES
(145,'Yes','Yes','Small');
INSERT INTO Dogs(ID,DogFriendly,ChildFriendly,Size) VALUES
(146,'Yes','No','Large');
INSERT INTO Dogs(ID,DogFriendly,ChildFriendly,Size) VALUES
(148,'Yes','Yes','Large');

```

- **employees.sql**

```

CREATE TABLE Employees(
    employee_id INTEGER NOT NULL PRIMARY KEY
    ,phone      INTEGER NOT NULL
    ,age        INTEGER NOT NULL
    ,salary     INTEGER NOT NULL
);

```

```

INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(0,5046218927,39,44000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(1,8102929388,34,39000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(2,8566368749,36,41000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(3,9073854412,23,28000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(4,5135701893,28,33000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(5,4195032484,24,29000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(6,7735736914,33,38000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(7,4087523500,25,30000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(8,6054142147,25,30000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(9,4106558723,30,35000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(10,2158741229,38,43000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(11,6313353414,33,38000);

```

```

INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(12,3104985651,22,27000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(13,4407808425,28,33000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(14,9565376195,25,30000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(15,6022774385,31,36000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(16,9313139635,30,35000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(17,4146619598,30,35000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(18,3132887937,26,31000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(19,8158282147,38,43000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(20,6105453615,28,33000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(21,4085401785,37,42000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(22,9723039197,21,26000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(23,5189667987,26,31000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(24,7326583154,26,31000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(25,7156626764,36,41000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(26,9133882079,21,26000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(27,4106691642,32,37000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(28,2125824976,27,32000);
INSERT INTO Employees(employee_id,phone,age,salary) VALUES
(29,9363363951,35,40000);

```

- **users.sql**

```

CREATE TABLE Users(

```

```

    phone BIGINT NOT NULL PRIMARY KEY
    ,address VARCHAR(40) NOT NULL
    ,city VARCHAR(17) NOT NULL
    ,county VARCHAR(20) NOT NULL
    ,state VARCHAR(2) NOT NULL
    ,zip INTEGER NOT NULL
);

```

```

INSERT INTO Users(phone,address,city,county,state,zip) VALUES
(6148019788,'74874 Atlantic Ave','Columbus','Franklin','OH',43215);
INSERT INTO Users(phone,address,city,county,state,zip) VALUES
(5059773911,'366 South Dr','Las Cruces','Dona Ana','NM',88011);
INSERT INTO Users(phone,address,city,county,state,zip) VALUES
(2017096245,'45 E Liberty St','Ridgefield Park','Bergen','NJ',7660);

```

•
. check users.sql for all add data commands into the Users table

```

INSERT INTO Users(phone,address,city,county,state,zip) VALUES
(9733542040,'47565 W Grand Ave','Newark','Essex','NJ',7105);
INSERT INTO Users(phone,address,city,county,state,zip) VALUES
(7734465569,'4284 Dorigo Ln','Chicago','Cook','IL',60647);
INSERT INTO Users(phone,address,city,county,state,zip) VALUES
(9739273447,'6794 Lake Dr E','Newark','Essex','NJ',7104);

```

- **animalphoto.sql**

```

CREATE TABLE animal_photo(
    animals_id INTEGER NOT NULL
    ,photo_id INTEGER NOT NULL
    ,PRIMARY KEY(animals_id,photo_id)
);

```

```

INSERT INTO animal_photo(animals_id,photo_id) VALUES (1,1);
INSERT INTO animal_photo(animals_id,photo_id) VALUES (2,2);
INSERT INTO animal_photo(animals_id,photo_id) VALUES (3,3);

```

•
. check animalphoto.sql for all add data commands into the animal_photo table

```

INSERT INTO animal_photo(animals_id,photo_id) VALUES (146,146);
INSERT INTO animal_photo(animals_id,photo_id) VALUES (147,147);
INSERT INTO animal_photo(animals_id,photo_id) VALUES (148,148);

```

- **oversee.sql**

```
CREATE TABLE oversee(  
    employee_id INTEGER NOT NULL  
    ,animal_id  INTEGER NOT NULL PRIMARY KEY  
    ,start_date DATE NOT NULL  
    ,end_date   DATE NOT NULL  
);
```

```
INSERT INTO oversee(employee_id,animal_id,start_date,end_date) VALUES  
(0,1,'2021-12-13','2021-12-27');
```

```
INSERT INTO oversee(employee_id,animal_id,start_date,end_date) VALUES  
(0,2,'2022-04-17','2022-05-01');
```

```
INSERT INTO oversee(employee_id,animal_id,start_date,end_date) VALUES  
(0,3,'2022-05-06','2022-05-20');
```

.

. *check oversee.sql for all add data commands into the oversee table*

.

```
INSERT INTO oversee(employee_id,animal_id,start_date,end_date) VALUES  
(29,146,'2022-01-23','2022-02-06');
```

```
INSERT INTO oversee(employee_id,animal_id,start_date,end_date) VALUES  
(29,147,'2022-05-16','2022-05-30');
```

```
INSERT INTO oversee(employee_id,animal_id,start_date,end_date) VALUES  
(29,148,'2022-02-22','2022-03-08');
```

- **adopted.sql**

```
CREATE TABLE adopted(  
    animal_id          INTEGER NOT NULL PRIMARY KEY  
    ,phone             VARCHAR(14)  
    ,date_of_adoption  DATE  
    ,AdoptionStatus    VARCHAR(6) NOT NULL  
);
```

```
INSERT INTO adopted(animal_id,phone,date_of_adoption,AdoptionStatus)  
VALUES (1,'2017096245','2021-01-07','Foster');
```

```
INSERT INTO adopted(animal_id,phone,date_of_adoption,AdoptionStatus)  
VALUES (2,'2088625339','2021-02-25','Foster');
```

```
INSERT INTO adopted(animal_id,phone,date_of_adoption,AdoptionStatus)  
VALUES (3,'2108129597','2021-03-14','Foster');
```

- *. check adopted.sql for all add data commands into the adopted table*
- ```

INSERT INTO adopted(animal_id,phone,date_of_adoption,AdoptionStatus)
VALUES (146,NULL,NULL,'Foster');
INSERT INTO adopted(animal_id,phone,date_of_adoption,AdoptionStatus)
VALUES (147,NULL,NULL,'Foster');
INSERT INTO adopted(animal_id,phone,date_of_adoption,AdoptionStatus)
VALUES (148,NULL,NULL,'Adopt');

```

### **RETRIEVE DATA:**

- **Allow users to retrieve data from the database**

- Allow users to see what animals are available to adopt

```

SELECT AdoptionStatus FROM `animals` WHERE AdoptionStatus =
"Adopt";

```

- Allow users to see what animals are available to foster

```

SELECT AdoptionStatus FROM `animals` WHERE AdoptionStatus =
"Foster";

```

### **ADD DATA:**

- **Allow users to add data to the database**

- Allow users to add user information to their profile page (when they first create an account)
  - Name, profile picture, age, location, housing situation

```

INSERT INTO Accounts(phone,first_name,last_name,photo_id,hashed_pass)
VALUES (<phone>, <first_name>, <last_name>, <photo_id>, <hashed_pass>);

```

- Allow employees (higher access users) to put animals up for adoption

```

INSERT INTO
Animals(ID,Name,BirthDate,AnimalType,Sex,Breed,Description,AdoptionStatus)
VALUES(<id>, <name>, <BirthDate>, <AnimalType>, <Sex>, <Breed>, <Description>,
<AdoptionStatus>);

```



### **UPDATE DATA:**

- **Allow users to update data in the database**

- Shelter employees should be able to update attributes about an animal
  - e.g., available for foster, adoption, or not at all (undergoing surgery).

```
UPDATE animals
SET animals.<attribute> = [new value]
WHERE animals.id = <id of animal to update>
```

- Users can update information on their profile

```
UPDATE users
SET users.<attribute> = [new value]
WHERE users.phone = <phone number of user submitting the
update>
```

### **DELETE DATA:**

- **Allow users to delete data from the database**

- Shelter employees should be able to delete an animal from the database once the animal has been adopted.

```
DELETE FROM Animals WHERE animals.id = <id of animal that was
adopted>
```

### **ADVANCED COMMANDS:**

- **CHECK Constraint:**

```
CREATE TABLE Animals(
 ID INTEGER NOT NULL PRIMARY KEY
 ,Name VARCHAR(11)
 ,BirthDate DATE NOT NULL
 ,AnimalType VARCHAR(3) NOT NULL
 ,Sex VARCHAR(13) NOT NULL
 ,Breed VARCHAR(36) NOT NULL
 ,Description VARCHAR(19) NOT NULL
 ,AdoptionStatus VARCHAR(6) NOT NULL
 ,CONSTRAINT chk_ID CHECK (ID > 0)
);
```

- **Allow users to sort or filter the data**

- Users will be able to filter through breeds, ages, etc to filter the data.

```
SELECT * FROM Animals
WHERE Animals.<attribute> = [filter value]
```

Example for filtering animals for only dogs:

```
SELECT * FROM Animals
WHERE Animals.AnimalType = "Dog"
```

- Users will be able to sort the data by age, weight, etc

```
SELECT * FROM Animals
ORDER BY Animals.<attribute to sort by>
```

Example for sorting animals by date of birth with **STORED PROCEDURE**:

```
CREATE PROCEDURE sortByDOB()
SELECT * FROM animals
ORDER BY animals.BirthDate;
```

CALL sortByDOB(); # this can be called every time the user sorts the animals