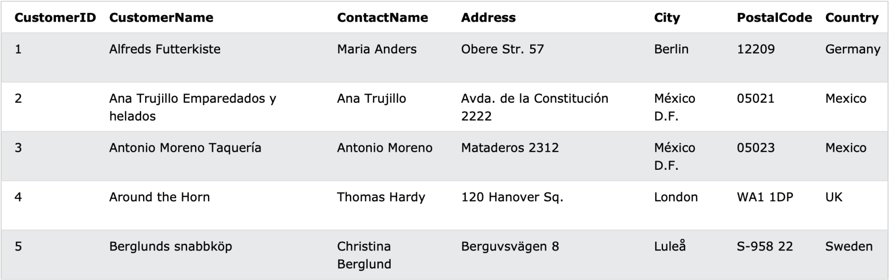
Practice 1



CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

1. Get all the columns from the `Customers` table

```
SELECT * FROM Customers
```

2. Write a statement that will select the City column from the Customers table.

```
SELECT City FROM Customers
```

3. Select all the different values from the Country column in the Customers table.

```
SELECT DISTINCT(Country) FROM Customers
```

4. Select all records where the City column has the value “Berlin”.

```
SELECT * FROM Customers WHERE City="Berlin"
```

5. Select all records where the CustomerID column has the value 3.

```
SELECT * FROM Customers WHERE CustomerID = 3
```

6. Select all records where the City column has the value ‘Berlin’ and the PostalCode column has the value ‘12209’.

```
SELECT * FROM Customers WHERE City = "Berlin" AND PostalCode = "12209"
```

7. Select all records where the City column has the value ‘Berlin’ or ‘London’.

```
SELECT * FROM Customers WHERE City = "Berlin" OR City = "London"
```

8. Select all records from the Customers table, sort the result alphabetically by the column City.

```
SELECT * FROM Customers ORDER BY City ASC;
```

9. Select all records from the Customers table, sort the result reversed alphabetically by the column City.

```
SELECT * FROM Customers ORDER BY City DESC;
```

10. Select all records from the Customers table, sort the result alphabetically, first by the column Country, then, by the column City.

```
SELECT * FROM Customers ORDER BY Country ASC, City ASC;
```

11. Select the record with the smallest value of the Price column.

```
SELECT MIN(Price) FROM Products;
```

12. Use an SQL function to select the record with the highest value of the Price column.

```
SELECT MAX(Price) FROM Products;
```

13. Return the number of records that have the Price value set to 18.

```
SELECT COUNT(*) FROM Products WHERE Price = 18;
```

14. Use an SQL function to calculate the average price of all products.

```
SELECT AVG(Price) FROM Products;
```

15. Use an SQL function to calculate the sum of all the Price column values in the Products table.

```
SELECT SUM(Price) FROM Products;
```

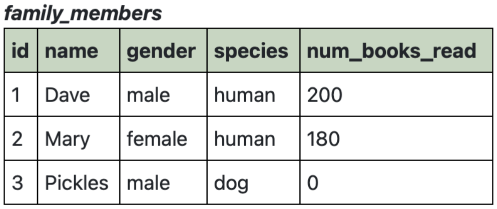
16. List the number of customers in each country.

```
SELECT Country, COUNT(*) FROM Customers GROUP BY Country;
```

17. List the number of customers in each country.

```
SELECT Country, COUNT(*) FROM Customers GROUP BY Country;
```

Practice 2



family_members

id	name	gender	species	num_books_read
1	Dave	male	human	200
2	Mary	female	human	180
3	Pickles	male	dog	0

1. Display all of that data in `family_members`

```
SELECT * FROM family_members
```

2. Display only the name and num_books_read columns

```
SELECT name, bum_books_read FROM family_members
```

- Return just the name and species columns

```
SELECT name, species FROM family_member
```

3. Grab all of the rows that correspond to humans

```
SELECT * FROM family_members WHERE species = "human
```

- Run a query that returns all of the rows that refer to dogs

```
SELECT * FROM family_members WHERE species = "dog"
```

4. Select family members that read at least 1 book.

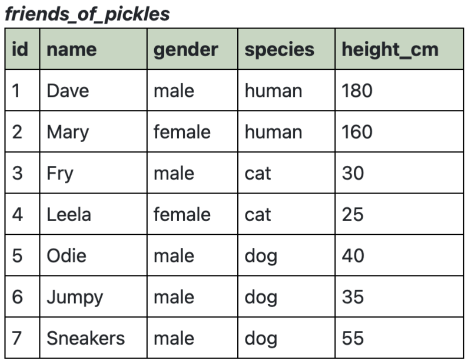
```
SELECT * FROM family_members WHERE num_books_read > 0
```

```
SELECT * FROM family_members WHERE num_books_read >= 1
```

- Return all rows of family members whose num_books_read is greater than 190

```
SELECT * FROM family_members WHERE num_books_read > 190
```

5. Return all rows in family_members where num_books_read is a value greater or equal to 180



friends_of_pickles

id	name	gender	species	height_cm
1	Dave	male	human	180
2	Mary	female	human	160
3	Fry	male	cat	30
4	Leela	female	cat	25
5	Odie	male	dog	40
6	Jumpy	male	dog	35
7	Sneakers	male	dog	55

1. Find the friends of Pickles that are over 25cm in height and are cats

```
SELECT * FROM friends_of_pickles WHERE height_cm > 25 AND species = 'cat';
```

- Find all of Pickles' friends that are dogs and under the height of 45cm

```
SELECT * FROM friends_of_pickles WHERE species = 'dog' AND height_cm < 45;
```

2. Find the friends of Pickles that are over 25cm in height or are cats

```
SELECT * FROM friends_of_pickles WHERE height_cm > 25 OR species = 'cat';
```

- Find all of Pickles' friends that are dogs or under the height of 45cm

```
SELECT * FROM friends_of_pickles WHERE species = 'dog' OR height_cm < 45;
```

3. Get the gender and species combinations of the animals less than 100cm in height.

```
SELECT gender, species FROM friends_of_pickles WHERE height_cm < 100;
```

- Return a list of the distinct species of animals greater than 50cm in height

```
SELECT DISTINCT(species) FROM friends_of_pickles WHERE height_cm > 50;
```

4. Sort the friends_of_pickles by name

```
SELECT * FROM friends_of_pickles ORDER BY name ASC;
```

- Run a query that sorts the friends_of_pickles by height_cm in descending order

```
SELECT * FROM friends_of_pickles ORDER BY height_cm DESC;
```

5. Sorts the friends_of_pickles by height_cm in descending order

```
SELECT * FROM friends_of_pickles ORDER BY height_cm DESC;
```

6. Return the total number of rows in the table

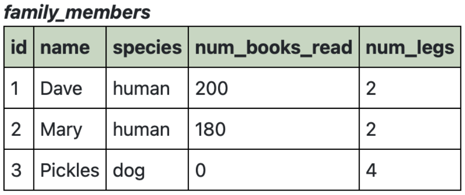
```
SELECT COUNT(*) FROM friends_of_pickles;
```

7. Return the total number of human friends of pickles

```
SELECT COUNT(*) FROM friends_of_pickles WHERE species = 'human';
```

- Return the number of rows in friends_of_pickles where the species is a dog

```
SELECT COUNT(*) FROM friends_of_pickles WHERE species = 'dog';
```



family_members

id	name	species	num_books_read	num_legs
1	Dave	human	200	2
2	Mary	human	180	2
3	Pickles	dog	0	4

1. Return the total number of legs in the family.

```
SELECT SUM(num_legs) FROM family_members;
```

- Find the total num__books__read made by this family

```
SELECT SUM(num_books_read) FROM family_members;
```

2. Returns the average number of legs of each family member.

```
SELECT AVG(num_legs) FROM family_members;
```

- Find the average num__books__read made by each family member

```
SELECT AVG(num_books_read) FROM family_members;
```

3. Find the least number of legs in a family member

```
SELECT MIN(num_legs) FROM family_members;
```

- Find the highest num__books__read that a family member makes

```
SELECT MAX(num_books_read) FROM family_members;
```

4. How many of each species does pickle has as a friend?

```
SELECT species, COUNT(*) FROM friends_of_pickles GROUP BY species;
```

- Return the tallest height for each species?

```
SELECT species, MAX(height_cm) FROM friends_of_pickles GROUP BY species;
```

References

Now click go to the following two website and practice the questions again!

w3schools.com

sql-esy.com