

✔ Congratulations! You passed!

Grade received **80%** Latest Submission Grade 80% To pass 80% or higher

[Go to next item](#)

Note- Some questions in this assessment require you to query in SQL on the FoodHunter Dataset. You can use the [Coursera Ungraded Labs](#)  to execute your queries.

1. Which statement is used to retrieve data from a database in MySQL?

1 / 1 point

- ☒ SELECT
- ☐ FROM
- ☐ COUNT
- ☐ DISTINCT



Correct

Correct! The SELECT statement is used to select data from a database.

2. Which keyword is used to specify the table from which data should be retrieved in a SELECT statement?

1 / 1 point

- ☐ SELECT
- ☐ COUNT
- ☒ FROM
- ☐ DISTINCT



Correct

Correct! The FROM command is used to specify which table to select or delete data from.

3. What is the purpose of the COUNT function in MySQL?

1 / 1 point

- ☐ It counts the total number of tables in a database
- ☒ It counts the total number of rows in a table.
- ☐ It counts the total number of columns in a table
- ☐ It counts the total number of unique values in a column.



Correct

Correct! You have understood the concept well.

4. Which keyword is used to retrieve only unique/distinct entries from a column in MySQL?

1 / 1 point

- ☐ UNIQUE

- ☐ FROM
- ☐ COUNT
- ☒ DISTINCT



Correct

Correct! The DISTINCT statement is used to return only distinct (different) values.

5.

1 / 1 point



Refer to the schema and convert the following text into an SQL query-

“Show all the data in the **restaurants** table”

Note- The case(lowercase, uppercase) for SQL keywords does not matter. Also, ensure you enter a semi-colon(;) at the end of the query (just like we have shown in the videos)

```
SELECT * FROM restaurants;
```

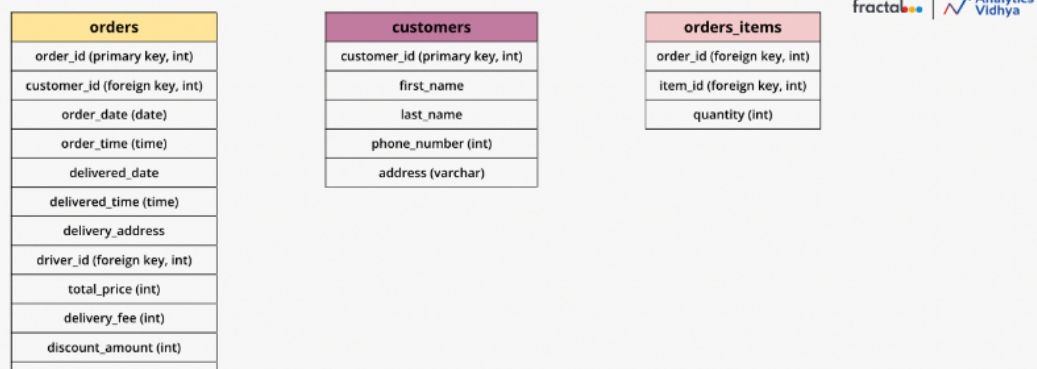


Correct

Congratulations! Your query is correct.

6.

0 / 1 point



total_price(float)
order_rating

restaurants
restaurant_id (primary key, int)
restaurant_name
cuisine
address

food_items
item_id (primary key, int)
restaurant_id (foreign key, int)
item_name (varchar)
food_type
price (int)
calories (int)

drivers
driver_id (foreign key, int)
name
rating

Relational Database
FoodHunter App

Refer to the schema and convert the following text into an SQL query-

“Show only the dishes, their prices, and the calories in them from the food items table”. Here **item_name = dishes**

Note- The case(lowercase, uppcase) for SQL keywords does not matter. Also, ensure you enter a semi-colon(;) at the end of the query (just like we have shown in the videos)

```
SELECT item_name AS 'dishes',price,calories FROM food_items;
```

✗ **Incorrect**

Your answer is incorrect! Refer to "Writing Your First Code to Explore ORDERS Table - Part 1" in Lesson 1 to understand this.

7.

0 / 1 point

orders
order_id (primary key, int)
customer_id (foreign key, int)
order_date (date)
order_time (time)
delivered_date
delivered_time (time)
delivery_address
driver_id (foreign key, int)
total_price (int)
delivery_fee (int)
discount_amount (int)
final_price(float)
order_rating

customers
customer_id (primary key, int)
first_name
last_name
phone_number (int)
address (varchar)

orders_items
order_id (foreign key, int)
item_id (foreign key, int)
quantity (int)

restaurants
restaurant_id (primary key, int)
restaurant_name
cuisine
address

food_items
item_id (primary key, int)
restaurant_id (foreign key, int)
item_name (varchar)
food_type
price (int)
calories (int)

drivers
driver_id (foreign key, int)
name
rating

Relational Database
FoodHunter App

Refer to the schema and write an SQL query to retrieve the order ids, customer ids, and total prices of all orders.

Note- The case(lowercase, uppcase) for SQL keywords does not matter. Also, ensure you enter a semi-colon(;) at the end of the query (just like we have shown in the videos)

```
SELECT order_id,customer_id,total_price FROM orders ;
```

✗ **Incorrect**

Your answer is incorrect! Refer to "Writing Your First Code to Explore ORDERS Table - Part 1" in Lesson 1 to understand this.

8.

1 / 1 point

orders

order_id (primary key, int)

customer_id (foreign key, int)

order_date (date)

order_time (time)

delivered_date

delivered_time (time)

delivery_address

driver_id (foreign key, int)

total_price (int)

delivery_fee (int)

discount_amount (int)

final_price(float)

order_rating

restaurants

restaurant_id (primary key, int)

restaurant_name

cuisine

address

customers

customer_id (primary key, int)

first_name

last_name

phone_number (int)

address (varchar)

orders_items

order_id (foreign key, int)

item_id (foreign key, int)

quantity (int)

drivers

driver_id (foreign key, int)

name

rating

food_items

item_id (primary key, int)

restaurant_id (foreign key, int)

item_name (varchar)

food_type

price (int)

calories (int)

fractal

Vidhya

Relational Database

FoodHunter App

Refer to the schema and write a query to count the number of **restaurants** in the "restaurants" table using restaurant id.

Note- The case(lowercase/uppercase) for SQL keywords does not matter. Also, ensure you enter a semi-colon (;) at the end of the query (just like we have shown in the videos).

```
SELECT COUNT(restaurant_id) FROM restaurants;
```

✓ **Correct**

Congratulations! Your query is correct.

9.

1 / 1 point

orders

order_id (primary key, int)

customer_id (foreign key, int)

order_date (date)

order_time (time)

delivered_date

delivered_time (time)

delivery_address

driver_id (foreign key, int)

total_price (int)

delivery_fee (int)

discount_amount (int)

final_price(float)

order_rating

customers

customer_id (primary key, int)

first_name

last_name

phone_number (int)

address (varchar)

orders_items

order_id (foreign key, int)

item_id (foreign key, int)

quantity (int)

restaurants

restaurant_id (primary key, int)

restaurant_name

cuisine

address

food_items

item_id (primary key, int)

restaurant_id (foreign key, int)

item_name (varchar)

food_type

price (int)

calories (int)

drivers

driver_id (foreign key, int)

name

rating

fractal

Vidhya

Relational Database

FoodHunter App

Refer to the schema and write a code to find the unique number of cuisines served by the restaurants from the **restaurants** table.

Note- The case(lowercase/uppercase) for SQL keywords does not matter. Also, ensure you enter a semi-colon (;) at the end of the query (just like we have shown in the videos).

```
SELECT COUNT(DISTINCT cuisine) FROM restaurants;
```

✓ **Correct**

Congratulations! Your query is correct.

10.

1 / 1 point

orders
order_id (primary key, int)
customer_id (foreign key, int)
order_date (date)
order_time (time)
delivered_date
delivered_time (time)
delivery_address
driver_id (foreign key, int)
total_price (int)
delivery_fee (int)
discount_amount (int)
final_price(float)
order_rating

restaurants
restaurant_id (primary key, int)
restaurant_name
cuisine
address

customers
customer_id (primary key, int)
first_name
last_name
phone_number (int)
address (varchar)

food_items
item_id (primary key, int)
restaurant_id (foreign key, int)
item_name (varchar)
food_type
price (int)
calories (int)

orders_items
order_id (foreign key, int)
item_id (foreign key, int)
quantity (int)

drivers
driver_id (foreign key, int)
name
rating

fractal | Analytics Vidhya

Relational Database
FoodHunter App

Refer to the schema and write a code to find the number of **unique dishes** served by **restaurants** from the **food_items** table using the item_name.

Note- The case(lowercase, uppercase) for SQL keywords does not matter. Also, ensure you enter a semi-colon(;) at the end of the query (just like we have shown in the videos)

```
SELECT COUNT(DISTINCT item_name) FROM food_items;
```

✓ **Correct**

Congratulations! Your query is correct.