



✓ **Congratulations! You passed!**
TO PASS 80% or higher

Keep Learning

GRADE
100%

Weekly challenge 4

LATEST SUBMISSION GRADE

100%

1. A data analyst is working with a spreadsheet from a furniture company. To use the template for this spreadsheet, click the link below and select "Use Template."

1 / 1 point

Link to template: [Sample Transaction Table](#)

Or, if you don't have a Google account, download the file directly from the attachment below.



Sample Transaction Table - transactional-data-format-csv
CSV File

[Download file](#)

The analyst inputs a function to find the number of product prices that are less than \$150.00. Which formula will return that result?

- ☐ =SUMIF(G2:G30, "<150")
- ☐ =COUNTIF(G2:G30, ">=150")
- ☒ =COUNTIF(G2:G30, "<150")
- ☐ =SUMIF(G2:G30, ">150")



Correct

The COUNTIF formula =COUNTIF(G2:G30, "<150") will allow the analyst to count all product price values in Column G that are less than \$150.

2. Fill in the blank: When you build a SUMIF or COUNTIF formula, the first part of the formula in parentheses is the ____.

1 / 1 point

- ☐ condition
- ☐ criteria
- ☒ range
- ☐ operator



Correct

When you build a SUMIF or COUNTIF formula, the first part of the formula in parentheses is the range. The range of cells is then evaluated by the criteria or condition that you include in the formula.

3. The following is a formula with the SUMPRODUCT function:

1 / 1 point

=SUMPRODUCT(A2:A10,B2:B10).

It will add the values from the first range (A2:A10) to the values from the second range (B2:B10). Then, the sums will be multiplied.

- ☐ True
- ☒ False



Correct

The formula will multiply the first range of values (A2:A10) by the second range of values (B2:B10). Then, the products will be added together.

4. You create a pivot table in a spreadsheet containing movie data. To use the template for this spreadsheet, click the link below and select "Use Template."

1 / 1 point

Link to template: [Movie Data Project](#).

Or, if you don't have a Google account, download the file directly from the attachment below.



Movie Data Starter Project
XLSX File

[Download file](#)

In order to find out how much box office revenue each genre earned, you would use the SUM function in the Values menu to summarize the data.

- ☒ True
- ☐ False

✓ **Correct**

You would use the SUM function to figure out how much box office revenue each genre earned. In the pivot table, the SUM function would add the total revenue separately for each genre.

5. A data analyst uses the following SQL query to perform basic calculations on their data. Which types of operators is the analyst using in this SQL query? Select all that apply.

1 / 1 point

```
SELECT
Yes_Responses,
No_Responses,
Total_Surveys,
(Yes_Responses + No_Responses) / Total_Surveys AS Responses_Per_Survey
FROM
Survey_1
```

- ☒ Division

✓ **Correct**

The analyst is using the division operator (/) in this SQL query to divide the sum of "yes" and "no" responses by the total number of surveys.

- ☒ Addition

✓ **Correct**

The analyst is using the addition operator (+) in this SQL query to calculate the sum of "yes" and "no" responses.

- ☐ Subtraction

- ☐ Multiplication

6. You are working with a database table that contains data about music. The table includes columns for *track_id*, *track_name* (name of the music track), *composer*, and *bytes* (digital storage size of the music track). You are only interested in data about the classical musician Johann Sebastian Bach. You want to know the size of each Bach track in kilobytes. You decide to divide bytes by 1000 to get the size in kilobytes, and use the AS command to store the result in a new column called *kilobytes*.

1 / 1 point

Add a statement to your SQL query that calculates the size in kilobytes for each track and stores it in a new column as *kilobytes*.

NOTE: The three dots (...) indicate where to add the statement.

```
1  SELECT
2  track_id,
3  track_name,
4  composer,
5  bytes/1000 as Kilobytes
6  FROM
7  track
8  WHERE
9  composer = "Johann Sebastian Bach"
```

Run

Reset

What is the size in kilobytes of the track with Id number 3407?

- ☐ 4744
- ☒ 3192
- ☐ 5064
- ☐ 2315

✓ **Correct**

You add the statement `bytes / 1000 AS kilobytes` to calculate the size in kilobytes for each track and store it in a new column as kilobytes. The complete query is `SELECT track_id, track_name, composer, bytes, bytes / 1000 AS kilobytes FROM track WHERE composer = "Johann Sebastian Bach"`. The AS command gives a temporary name to the new column.

The size of the track with Id number 3407 is 3192 kilobytes.

7. You are working with a database table that contains invoice data. The table includes columns for *customer_id* and *total* (total amount billed for each invoice). Some customers have multiple invoices. You want to find out the total amount billed to each customer, and store the result in a new column as *total_amount*.

1 / 1 point

You write the SQL query below. Add a GROUP BY clause that will group the data by customer ID number.

```
1 SELECT
2 customer_id,
3 SUM(total) AS total_amount
4 FROM
5 invoice
6 GROUP BY
7 customer_id
```

Run

Reset

What is the total amount billed to the customer with Id number 5?

- ☐ 37.62
- ☐ 39.62
- ☒ 40.62
- ☐ 49.62

✓ Correct

You add the clause `GROUP BY customer_id` to group the data by customer Id number. The complete query is `SELECT customer_id, SUM(total) AS total_amount FROM invoice GROUP BY customer_id`. The GROUP BY command groups rows that have the same values from a table into summary rows. GROUP BY is always placed as the last command in a SELECT-FROM-WHERE query.

The total amount billed to the customer with Id number 5 is 40.62.

8. You are working with a database table that contains invoice data. The table includes columns for *billing_state*, *billing_country*, and *total*. You want to know the average total price for the invoices billed to the state of Wisconsin. You decide to use the AVG function to find the average total, and use the AS command to store the result in a new column called *average_total*.

1 / 1 point

Add a statement to your SQL query that calculates the average total and stores it in a new column as *average_total*.

NOTE: The three dots (...) indicate where to add the statement.

```
1 SELECT
2 billing_state,
3 billing_country,
4 avg(total) AS average_total
5 FROM
6 invoice
7 WHERE
8 billing_state = "WI"
```

Run

Reset

What is the average total for Wisconsin?

- ☐ 5.37
- ☒ 6.08
- ☐ 5.54
- ☐ 5.78

✓ Correct

You add the statement `AVG(total) AS average_total` to calculate the average total and store it in a new column as *average_total*. The complete query is `SELECT billing_state, billing_country, AVG(total) AS average_total FROM invoice WHERE billing_state = "WI"`. The AVG function is an aggregate function that returns the average value of a group of values. The AS command gives a temporary name to the new column.

The average total for Wisconsin is 6.08.