1.

Question 1

In the first task, you used MySQL’s built-in concatenation function to combine two strings. How does MySQL concatenation (CONCAT) function work with the data on integer columns using Python?

1 / 1 point

The concatenation function subtracts the given numbers and returns a result as a string to Python.

The concatenation function adds the given data and Python converts it to a string when it returns the result.

The concatenation function combines the given integers as a string. Python only executes the query.

The concatenation function adds the given data and returns the sum as an integer to Python.

Correct

Correct. The concatenation function combines the integers as they are taken as strings to the function. For example, the concatenation of 12 and 34 is ‘1234’.

2.

Question 2

In the second task, what MySQL built-in functions did you use in your SQL query? Select all that apply.

1 / 1 point

AVG

Correct

Correct! AVG is a built-in MySQL function that returns the mean or average of the given numbers.

SUM

Correct

Correct! SUM is a built-in MySQL function that returns the sum of the given numbers.

SELECT

COUNT

Correct

Correct! COUNT is a built-in MySQL function that returns the number of records.

3.

Question 3

In the third task, you used MySQL built-in function COUNT along with the GROUP BY clause on the retrieved data. How can you combine MySQL functions with GROUP BY clause using Python?

1 / 1 point

Embed the MySQL functions and the GROUP BY clause in a SQL query.

Access MySQL function in Python outside the SQL query and make the GROUP BY clause part of your SQL query.

Embed the MySQL functions in Python code and run it with a SQL query that contains the GROUP BY clauses.

Use both the MySQL function and the GROUP BY clause in your Python code.

Correct

Correct! MySQL function COUNT and the GROUP BY clause are part of your SQL query. Python executes that query via MySQL Connector/Python.

4.

Question 4

In the fourth task, you used the CASE function from MySQL to implement a conditional block. While using Python, you separated each condition using a default delimiter in the stored procedure as Python does not support multiple conditions in a single CASE function.

1 / 1 point

True

False

Correct

Correct! You can use multiple conditions in a CASE function. They don’t need to be separated by the delimiter. Python does not interfere with the SQL syntax.