Congratulations! You passed!
TO PASS 80% or higher

Keep Learning

GRADE
100%

Graded Quiz: Test your knowledge about relational database and SQL

LATEST SUBMISSION GRADE
100%

1. Which of the following statements is **FALSE** about relational databases? — 1 / 1 point

- Tables are major components of a relational database and they hold the data.
- Rows in tables can be visualized as "records"
- Tables in a relational database are always independent from each other.
- Tables can be visualized as having columns and rows.

Correct
This statement is incorrect and therefore is the right answer. Tables often relate to each other by common data in columns.

2. Which of the following is a valid SQL statement to select all the members who live in Texas from the table **club_member**? — 1 / 1 point

- SELECT * FROM **club_member** WHERE state_abbreviation = "TX"
- SELECT * FROM **club_member** WHERE state_abbreviation is TX
- SELECT * FROM **club_member**
- SELECT * FROM **club_member** WHERE state_abbreviation = TX

Correct
Correct! The syntax is SELECT [columns] FROM [table_name] WHERE [criteria]

3. Which of the following is a valid SQL statement to select the last_name and first_name columns of the **club_member** table? — 1 / 1 point

- SELECT **club_member** last_name, first_name
- SELECT **club_member** first_name and last_name
- SELECT last_name AND first_name FROM **club_member**
- SELECT last_name, first_name FROM **club_member**

Correct
Correct! Specify the columns you want to include after the keyword SELECT and use commas to separate them if there are more than one column.

4. Which of the following should be added to the end of this SQL statement to sort our query results by professional_title? — 1 / 1 point

SELECT * FROM **club_member** WHERE city = "San Francisco" _____

- IN professional_title ORDER
- SORT = professional_title
- SORT BY professional_title

- ◉ ORDER BY professional_title

✓ **Correct**
Correct!

5. Which of the following is a valid SQL statement to insert a row for a member named "Shino Yamaha" with a phone_number "4151234567" into the table **club_member**? **1 / 1 point**

- ○ INSERT INTO **club_member** SET first_name : "Shino", last_name : "Yamaha", phone_number : "4151234567"
- ○ INSERT "Shino, Yamaha, 4151234567" INTO **club_member**
- ○ INSERT NEW **club_member** SET first_name = "Shino", last_name = "Yamaha", phone_number = "4151234567"
- ◉ INSERT INTO **club_member** SET first_name = "Shino", last_name = "Yamaha", phone_number = "4151234567"

✓ **Correct**
Correct! The syntax is INSERT INTO [table] SET [column = value], [column = value] ...

6. What does this SQL statement do? **1 / 1 point**

SELECT first_name, last_name

FROM **employee**

ORDER BY last_name DESC

- ◉ It selects the columns first_name, last_name from the table **employee** and sort the results by last_name in descending order.
- ○ It selects the columns first_name, last_name from the table **employee** and sort the results by last_name.
- ○ An error will occur because this is not a valid SQL statement.
- ○ It selects the columns first_name, last_name from the table **employee** and sort the results by last_name description.

✓ **Correct**
Correct!

7. Which of these is a valid SQL statement to delete all the rows in the table model where the column type has a value "X"? **1 / 1 point**

- ○ DELETE * FROM model WHERE type = "X"
- ○ DELETE FROM model IF type = X
- ○ DELETE type X from model
- ◉ DELETE FROM model WHERE type = "X"

✓ **Correct**
Correct! The syntax is DELETE FROM [table] WHERE [criteria]

8. When this SQL statement is executed, what will happen? **1 / 1 point**

SELECT first_name, last_name, amount

FROM **club_member**, **invoice**

WHERE **club_member**.member_id = **invoice**.member_id

- It will join the tables **club_member** and **invoice** by member_id and list all the first_name, last_name and amounts of all the members who have invoices. But a member's first_name and last_name will be listed once if he/she has more than one invoice.
- It will produce an error because you cannot select rows from more than one table.
- It will join the tables **club_member** and **invoice** by member_id and list all the first_name, last_name and amounts of all the members who have invoices. A member's first_name and last_name will be repeated in each invoice if he/she has more than one.
- It will select all the members and list their first_name, last_name.

✓ **Correct**
Correct!

9. Which of the following is a valid SQL statement to link the **club_member** table to the **tier** table using tier_code and display each member's first_name, last_name and **tier**.description? **1 / 1 point**

- SELECT first_name, last_name, tier.description
  FROM **club_member**, **tier**
  WHERE **club_member**.tier_code = **tier**.code
- SELECT first_name, last_name, **tier**.description
  FROM **tier**
  WHERE **club_member**.tier_code = **tier**.code
- SELECT first_name, last_name, **tier**.description
  FROM **club_member**
  WHERE **club_member**.tier_code = **tier**.code
- SELECT first_name, last_name, **tier**.description
  FROM **club_member**<=>**tier**
  WHERE **club_member**.tier_code<=>**tier**.code

✓ **Correct**
Correct!

10. Which of these SQL statements will select all the members with last names that start with the capital letter "M" in the table **club_member**? **1 / 1 point**

- SELECT * FROM **club_member** WHERE last_name LIKE "M%"
- SELECT * FROM **club_member** WHERE last_name = "M"
- SELECT * FROM **club_member** WHERE last_name = "M-"
- SELECT * FROM **club_member** WHERE last_name = "M%"

✓ **Correct**
Correct. When you use the keyword LIKE, the percent sign "%" acts as a wildcard character.