Documentation: Combining Two Tables in SQL

Objective

The goal is to retrieve a list of persons with their first and last names along with their corresponding city and state from the Person and Address tables. If a person does not have an address listed in the Address table, the city and state should be reported as NULL.

Tables Description

1. Person Table

- Contains basic information about individuals.
- ***Columns:***
  - *personId (int):* Unique identifier for each person. This is the primary key.
  - *lastName (varchar):* The last name of the person.
  - *firstName (varchar):* The first name of the person.

2. Address Table

- Contains address details associated with individuals.
- ***Columns:***
  - *addressId (int):* Unique identifier for each address. This is the primary key.
  - *personId (int):* Foreign key that links to the personId in the Person table.
  - *city (varchar):* The city where the person resides.
  - *state (varchar):* The state where the person resides.

Requirements

- **Join the Person and Address tables:** The task requires combining data from both tables using the personId as the common field.
- **Return all persons from the Person table:** Every person listed in the Person table should be included in the result set, regardless of whether they have a corresponding entry in the Address table.
- **Include NULL for missing address information:** If a personId does not have a matching entry in the Address table, the output should display NULL for the city and state.

Approach

To achieve the desired output, the solution involves using the SQL LEFT JOIN clause:

1. **Selecting Data:** Start by selecting the firstName and lastName from the Person table. Then, select the city and state from the Address table.

2. **Using LEFT JOIN:**

- The LEFT JOIN clause is used to ensure that all records from the Person table are included in the result set, even if there are no matching entries in the Address table.
- This type of join returns all rows from the left table (in this case, Person) and the matched rows from the right table (Address). If there is no match, the result is NULL on the side of the Address table.

3. **Matching Condition:** The ON clause specifies that the join condition is based on matching the personId from both tables.

4. **Result Set:**

- The output will contain four columns: firstName, lastName, city, and state.
- If a person does not have a corresponding address in the Address table, the city and state fields will display NULL.

Example

Consider the following example to illustrate the join:

Input Data:

Person table:

personId	lastName	firstName
1	Wang	Allen
2	Alice	Bob

Address table:

addressId	personId	city	state
1	2	New York City	New York
2	3	Leetcode	California

Expected Output:

firstName	lastName	city	state
Allen	Wang	Null	Null
Bob	Alice	New York City	New York

Explanation:

- personId 1 (Allen Wang) does not have a matching entry in the Address table, so city and state are NULL.
- personId 2 (Bob Alice) has a matching address in New York City, New York.

Conclusion

- By utilizing a LEFT JOIN, the SQL query ensures that every person from the Person table is listed, along with their corresponding city and state, if available. If a person does not have an associated address, the output will display NULL for the city and state fields, thereby fulfilling the requirements of the task.