

WEEK 7

Constructing a **Select** **Statement**

PRESENTED BY FILSAN MUSA & FADUMO DIRIYE



TABLE OF CONTENT

- Understand the SELECT and FROM clauses.
- Learn how to retrieve data using SQL syntax
- Explore the use of DISTINCT and ALIASES for efficient querying.
- Apply these concepts to real-world examples.
- Recap and Next Steps

Overview of SELECT Statement

What is SELECT?

- The fundamental clause for retrieving data.
- Pulls columns or manipulated data from specified tables.

Table: Customers

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

SELECT first_name, last_name
FROM Customers;

first_name	last_name
John	Doe
Robert	Luna
David	Robinson
John	Reinhardt
Betty	Doe

What is FROM?

- Identifies the table(s) to retrieve data from.
- Specifies the database for accurate table referencing.

Using SELECT: Retrieve All Columns

Table: Customers

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

SELECT *
FROM Customers;

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

Explanation:

- Retrieves all columns from a table.

Using SELECT: Retrieve Specific Columns

Syntax Example:

sql

```
SELECT column_1  
FROM table_a;
```

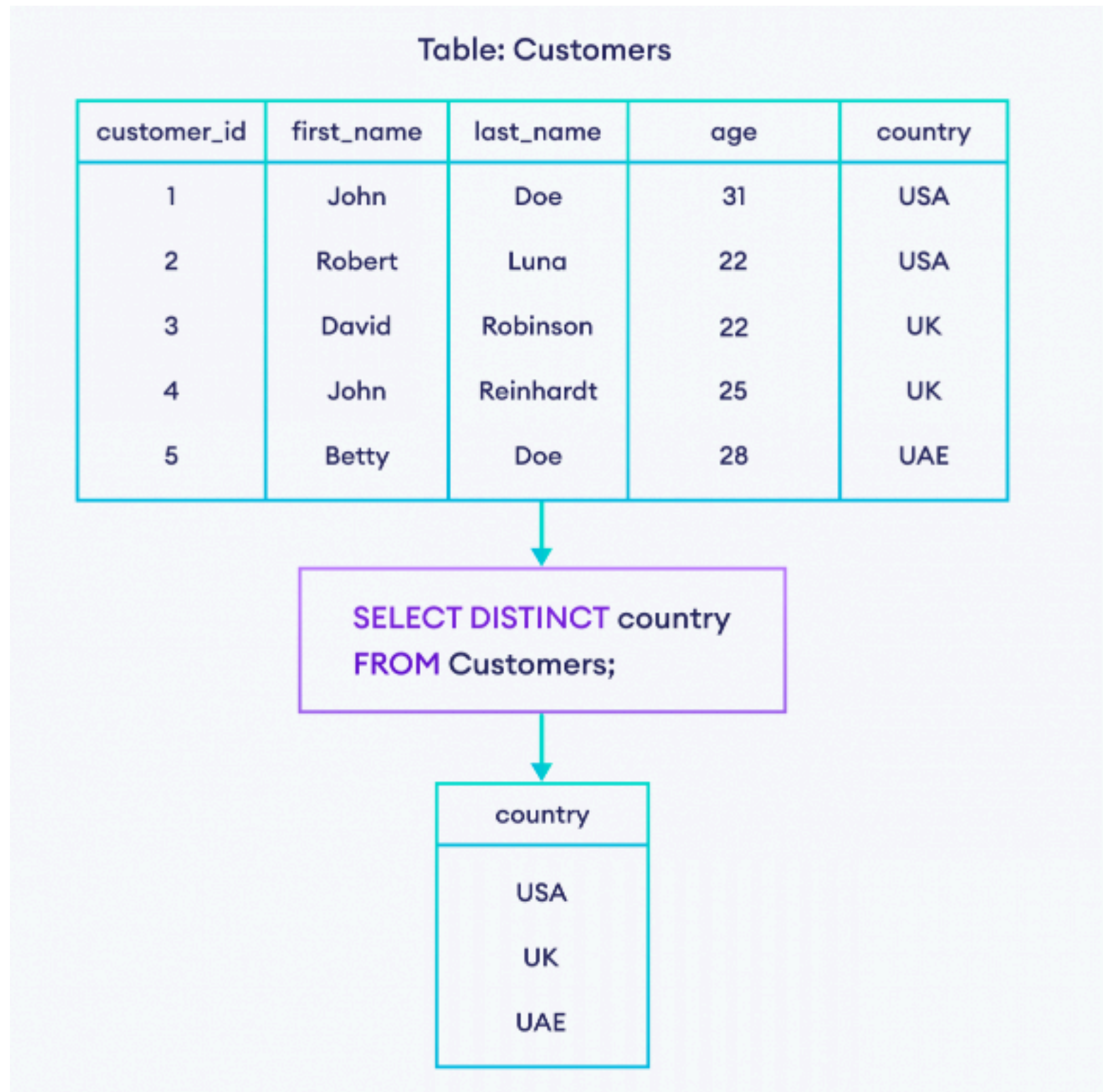
sql

```
SELECT column_1, column_2, column_3  
FROM table_a;
```

Explanation:

- Retrieves only the specified column(s)
- Specify multiple columns using commas

DISTINCT: Eliminating Duplicates



What is DISTINCT?

- Extracts unique values from a column.

ALIASES: Temporary Nicknames

Table: Customers

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

`SELECT first_name AS name
FROM Customers;`

name
John
Robert
David
John
Betty

What are ALIASES?

- Temporary names for columns and tables.
- Improve query readability and usability.

Note:

The “AS” keyword is optional.
Aliases can follow directly after the name.

Practical Application

Scenario: Retrieve unique product names and assign them a user-friendly alias.

SQL Example:

```
sql

SELECT DISTINCT product_name AS Product
FROM sales_data AS Sales;
```

Explanation:

- Retrieves unique product names and renames them for simplicity.

Recap and Next Steps

- Summary:
 - SELECT and FROM are the foundation of SQL queries.
 - Use DISTINCT to filter out duplicates.
 - ALIASES enhance readability and simplify complex queries.
 - Always tailor queries to your data needs.
- Next Steps:
 - Complete Week 7 Quiz