

SQL PREP

VERİ (Data): Raw information obtained through measurement, counting, experimentation, observation, or research.

VERİTABANI (Database): Systems that provide organized storage of data.

If we have organized data;

We can make forward-looking, developmental decisions.

We can solve our mistakes more easily.

We can make successful future predictions.

Now, let's address the critical question. For example, we can store our data in software similar to EXCEL. Why do we need a database?

Through databases, we ensure that the data in columns are of the same data type.

Databases allow us to work more easily with very large data sets.

Databases are more suitable for managing multiple users.

Databases work more easily with other software and applications.

What is SQL?

SQL stands for Structured Query Language.

SQL enables you to access and modify databases.

SQL became a standard in 1986 by the American National Standards Institute (ANSI) and in 1987 by the International Organization for Standardization (ISO).

What Can SQL Do?

SQL can execute queries against a database.

SQL can retrieve data from a database.

SQL can insert records into a database.

SQL can update records in a database.

SQL can delete records from a database.

SQL can create new databases.

SQL can create new tables in a database.

SQL can create procedures in a database.

SQL can create views in a database.

SQL can set permissions on tables, procedures, and views.

RDBMS

RDBMS stands for Relational Database Management System.

RDBMS is the foundation for all modern database systems like SQL, MS SQL Server, Oracle, MySQL, and Microsoft Access.

In RDBMS, data are stored in objects called tables within the database. A table is a collection of related data entries and consists of columns and rows.

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany
7	Blondel père et fils	Frédérique Citeaux	24, place Kléber	Strasbourg	67000	France
8	Bólido Comidas preparadas	Martin Sommer	C/ Araquil, 67	Madrid	28023	Spain
9	Bon app'	Laurence Leblhans	12, rue des Bouchers	Marseille	13008	France
10	Bottom-Dollar Marketse	Elizabeth Lincoln	23 Tsawassen Blvd.	Tsawassen	T2F 8M4	Canada

Each table is divided into smaller parts called fields.

The fields in the Customers table include CustomerID, RelatedPersonName, Address, City, Postal Code, and Country. A field is designed to keep specific information about each record in the table, which is a column in the designed table.

A record (row) is defined as each separate entry in a table.

For example, the Customers table contains 10 records. A record is a horizontal presence in the table.

A column is a vertical entity in the table that contains all the information related to a specific field.

PostgreSQL

PostgreSQL is a relational database management system (RDBMS). This means it is a system for managing data stored in relations.

PostgreSQL Data Types

PostgreSQL supports mainly nine numerical data types:

INTEGER, SMALLINT, BIGINT, DECIMAL, NUMERIC, SERIAL, BIGSERIAL, DOUBLE, REAL.

DATE and TIME data types are used for storing date and time information.

For storing fixed-length string values, the CHAR(n) type is used, and for variable-length strings, the VARCHAR(n) type is used.

Creating a Table

You can create a table by specifying the table name, all column names, and their types.

```
CREATE TABLE weather (  
  city      varchar(80),  
  temp_low  int,      -- düşük sıcaklık  
  temp_high int,      -- yüksek sıcaklık  
  prcp      real,     -- yağış  
  date      date  
):
```

If you no longer need a table or want to recreate it differently, to remove the table:

DROP TABLE tablename;

(Instruction to remove the table is implied but not visible in the text shown)

To add a row to a table:

INSERT INTO weather

VALUES ('San Francisco', 46, 50, 0.25, '1994-11-27');

(Instruction to add a row is implied but not visible in the text shown)

Querying a Table

To retrieve data from a table, table queries are used. This is done using the SQL SELECT statement. The statement is divided into a selection list (the part that lists the columns to be returned), a table list (the part that lists the tables from which data will be retrieved), and an optional condition (the part that specifies any restrictions). For example, to get all the rows from a table:

(The specific example or instruction appears to be cut off and is not visible in the text shown)

SELECT * FROM weather;

Frequently Used SQL Commands

CREATE DATABASE: Used to create a new SQL database.

DROP DATABASE: Used to delete an existing SQL database.

CREATE TABLE: Used to create a new table in a database.

DROP TABLE: Used to delete an existing table from a database.

Frequently Used SQL Commands

ALTER TABLE: Used to add, delete, or modify columns in an existing table.

SELECT: Used to select data from a database.

SELECT DISTINCT: Used to return only distinct (different) values.

WHERE: Used to filter records.

AND and OR: Used to filter records based on more than one condition:

AND: If all the separated conditions are TRUE, then the record is displayed.

OR: If any of the separated conditions is TRUE, then the record is displayed.

Frequently Used SQL Commands

ORDER BY: Used to sort the result set in ascending or descending order.

INSERT INTO: Used to insert new records into a table.

UPDATE: Used to modify existing records in a table.

DELETE: Used to delete existing records from a table.

Frequently Used SQL Commands

JOIN: Used to combine rows from two or more tables based on a related column between them.

UNION: Used to combine the result sets of two or more SELECT statements.

GROUP BY: Used in a SQL SELECT statement to arrange identical data into groups, based on one or more columns.

HAVING: Used to apply a condition to groups created by GROUP BY.

EXISTS: Used to test for the existence of any record in a subquery.

SQL Constraints (Constraints)

SQL constraints are used to specify rules for data in a table.

Constraints are used to limit the type of data that can enter a table. This ensures the accuracy and reliability of the data in the table. If any action violates a constraint, the action is aborted.

Constraints can be specified at the column level or table level. Column level constraints apply to a single column, whereas table level constraints apply to the entire table.

SQL Constraints (Constraints)

NOT NULL: Ensures that a column cannot have a NULL value.

UNIQUE: Ensures all values in a column are different.

PRIMARY KEY: A combination of NOT NULL and UNIQUE. Identifies each row in a table uniquely.

FOREIGN KEY: Ensures the integrity of the relationships between tables.

CHECK: Ensures that all values in a column satisfy a specific condition.

DEFAULT: Sets a default value for a column if no value is specified.