SQL Server: Unique Constraints

This SQL Server tutorial explains how to **create, add, and drop unique constraints** in SQL Server with syntax and examples.

What is a unique constraint in SQL Server?

A unique constraint is a single field or combination of fields that uniquely defines a record. Some of the fields can contain null values as long as the combination of values is unique.

What is the difference between a unique constraint and a primary key?

Primary Key	Unique Constraint
None of the fields that are part of the primary key can contain a null value.	Some of the fields that are part of the unique constraint can contain null values as long as the combination of values is unique.

Create unique Contraint - Using a CREATE TABLE statement

The syntax for creating a unique constraint using a CREATE TABLE statement in SQL Server is:

```
CREATE TABLE table_name
(
  column1 datatype [ NULL | NOT NULL ],
  column2 datatype [ NULL | NOT NULL ],
  ...

  CONSTRAINT constraint_name UNIQUE (uc_col1, uc_col2, ... uc_col_n)
);
```

table_name
The name of the table that you wish to create.

column1, column2
The columns that you wish to create in the table.

constraint_name
The name of the unique constraint.

uc_col1, uc_col2, ... uc_col_n
The columns that make up the unique constraint.

Example

Let's look at an example of how to create a unique constraint in SQL Server using the CREATE TABLE statement.

```
CREATE TABLE employees
( employee_id INT PRIMARY KEY,
  employee_number INT NOT NULL,
  last_name VARCHAR(50) NOT NULL,
  first_name VARCHAR(50),
  salary MONEY,
  CONSTRAINT employees_unique UNIQUE (employee_number)
);
```

In this example, we've created a unique constraint on the employees table called employees_unique. It consists of only one field which is the employee_number.

We could also create a unique constraint with more than one field as in the example below:

```
CREATE TABLE employees
( employee_id INT PRIMARY KEY,
  employee_number INT NOT NULL,
  last_name VARCHAR(50) NOT NULL,
  first_name VARCHAR(50),
  salary MONEY,
  CONSTRAINT employees_unique UNIQUE (last_name, first_name)
);
```

Create unique contraint - Using an ALTER TABLE statement

The syntax for creating a unique constraint using an ALTER TABLE statement in SQL Server is:

```
ALTER TABLE table_name
ADD CONSTRAINT constraint_name UNIQUE (column1, column2, ... column_n);
```

table_name
The name of the table to modify. This is the table that you wish to add a unique constraint to.

constraint_name
The name of the unique constraint.

column1, column2, ... column_n
The columns that make up the unique constraint.

Example

Let's look at an example of how to add a unique constraint to an existing table in SQL Server using the ALTER TABLE statement.

```
ALTER TABLE employees
ADD CONSTRAINT employees_unique UNIQUE (employee_number);
```

In this example, we've created a unique constraint on the existing employees table called employees_unique. It consists of the field called employee_number.

We could also create a unique constraint with more than one field as in the example below:

```
ALTER TABLE employees
ADD CONSTRAINT employee_name_unique UNIQUE (last_name, first_name);
```

Drop Unique Constraint

The syntax for dropping a unique constraint in SQL Server is:

```
ALTER TABLE table_name
DROP CONSTRAINT constraint_name;
```

table_name
The name of the table to modify. This is the table whose unique constraint you wish to remove.

constraint_name
The name of the unique constraint to remove.

Example

Let's look at an example of how to remove a unique constraint from a table in SQL Server.

```
ALTER TABLE employees
DROP CONSTRAINT employees_unique;
```

In this example, we're dropping a unique constraint on the employees table called employees_unique.