SQL DEFAULT on CREATE TABLE

The `DEFAULT` constraint in SQL is used to provide a default value for a column when no value is specified during the insertion of a new row.

Syntax:

You can add a `DEFAULT` constraint when creating a table using the `CREATE TABLE` statement.

```sql
CREATE TABLE table_name (
    column_name datatype DEFAULT default_value
);
```

Example:

```sql
CREATE TABLE Employees (
    EmployeeID INT,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    HireDate DATE DEFAULT GETDATE()  -- Default value is the current date
);
```

In this example:

- The `HireDate` column will automatically take the current date when a new row is inserted, if no value is provided for `HireDate`.

Key Points about the `DEFAULT` Constraint:

1. **Optional Value**: The `DEFAULT` constraint allows you to set a default value for a column when no value is provided during an insert.
2. **Constant or Expression**: The default value can be a constant or an expression. For example, `GETDATE()` for the current date or a numeric value like `0`.
3. **Works with INSERT**: If you do not specify a value for a column with a `DEFAULT` constraint when inserting data, SQL Server will use the default value.

Example with an Insert:

```sql
sql                                                    ⎘ Copy   ✎ Edit

INSERT INTO Employees (EmployeeID, FirstName, LastName)
VALUES (1, 'Sagar', 'Khemchandani');   -- No HireDate provided, so it will default to curre
```

After this insert:

- The `EmployeeID`, `FirstName`, and `LastName` will be provided values.

- The `HireDate` will be set to the current date automatically, as no value was supplied for it.

To change the default value of a column in SQL, you need to drop the existing `DEFAULT` constraint first and then add a new one with the desired value. Unfortunately, you cannot directly modify a `DEFAULT` value using the `ALTER COLUMN` statement.

Drop the Existing Default Constraint

Once you have the name of the existing default constraint (let's assume it's called `DF_HireDate`), you can drop it using the `ALTER TABLE` statement.

```sql
sql                                                    ⎘ Copy   ✎ Edit

ALTER TABLE Employees
DROP CONSTRAINT DF_HireDate;
```

Changing a Default Value:

You can modify the default value of a column using the `ALTER TABLE` statement:

```sql
sql                                                    ⎘ Copy   ✎ Edit

ALTER TABLE Employees
ADD CONSTRAINT DF_HireDate DEFAULT '2025-01-01' FOR HireDate;
```

In this example:

- The default value for the `HireDate` column is now set to `2025-01-01`.

Difference Between `DEFAULT` and `NULL`:

- `DEFAULT`: Used to provide a specified value automatically when no value is supplied.
- `NULL`: No value at all. If a column allows `NULL`, and no value is provided, it will be stored as `NULL`.

Example of Using `DEFAULT` **with** `NULL` :

```sql
                                                        Copy   Edit

CREATE TABLE Employees (
    EmployeeID INT,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100) DEFAULT NULL    -- Default is NULL if no value is provided
);
```

In this case, if you don't provide an email during the insertion, it will default to `NULL` .

Summary:

- **DEFAULT**: Assigns a default value to a column if no value is specified in an `INSERT` statement.
- It's very useful for setting default values like timestamps, flags, or initial values.

---

1. What is the `DEFAULT` constraint in SQL?

- **Answer**: The `DEFAULT` constraint is used to provide a default value for a column when no value is specified during the insertion of a new row.

2. Can you apply the `DEFAULT` constraint to multiple columns in a table?

- **Answer**: Yes, you can apply the `DEFAULT` constraint to multiple columns in a table. Each column can have its own default value, and if no value is provided during an `INSERT`, the default value is used.

3. Can the `DEFAULT` constraint be applied to a column after the table has already been created?

- **Answer**: Yes, you can add the `DEFAULT` constraint to an existing column using the `ALTER TABLE` statement. For example:

```sql
                                                        Copy   Edit

ALTER TABLE Employees
ADD CONSTRAINT DF_HireDate DEFAULT GETDATE() FOR HireDate;
```

4. What is the difference between a `DEFAULT` constraint and allowing a column to be `NULL`?

- **Answer**:

- o A `DEFAULT` constraint provides a predefined value when no value is specified during an insert.
- o A column that allows `NULL` means no value is stored if the column is not provided during the `INSERT`. The difference is that `DEFAULT` will insert a specific value, while `NULL` represents the absence of a value.

5. How does the `DEFAULT` constraint behave with `INSERT` statements?

- **Answer**: If a column has a `DEFAULT` constraint and no value is provided for that column during an `INSERT` statement, the default value is automatically inserted. For example:

```sql
INSERT INTO Employees (EmployeeID, FirstName, LastName)
VALUES (1, 'Sagar', 'Khemchandani');
```

If `HireDate` has a default value (e.g., `GETDATE()`), it will be automatically inserted.

6. Can you have a `DEFAULT` value for a `PRIMARY KEY` column?

- **Answer**: Generally, a `PRIMARY KEY` column is intended to uniquely identify each row, so you cannot assign a `DEFAULT` value that might violate this uniqueness. However, you could use default values like `NEWID()` for a unique identifier in SQL Server.

7. What are some limitations of the `DEFAULT` constraint?

- **Answer**:
  - o It cannot be applied to `TEXT`, `NTEXT`, or `IMAGE` data types in some versions of SQL.
  - o The `DEFAULT` value must be a constant or an expression, not a subquery or a value that changes for each row.

How do you remove the `DEFAULT` constraint from a column?

- **Answer**: You can drop the `DEFAULT` constraint using the `ALTER TABLE` statement. First, you need to find the constraint name (it is usually automatically generated) and then drop it:

```sql
ALTER TABLE Employees
DROP CONSTRAINT DF_HireDate;
```