IDENTITY in MS SQL Server (Auto-Increment)

The IDENTITY property in **MS SQL Server** is used to create an **auto-incrementing** column, typically for **primary keys**. It automatically generates a unique sequential number whenever a new row is inserted into the table.

Syntax

```sql
CREATE TABLE table_name (
    column_name INT IDENTITY(seed, increment) PRIMARY KEY,
    other_column datatype
);
```

- `seed` → The starting value of the identity column.
- `increment` → The value by which the identity column is increased.

1. Basic `IDENTITY` Example

```sql
CREATE TABLE Employees (
    EmployeeID INT IDENTITY(1,1) PRIMARY KEY,
    Name VARCHAR(100),
    Age INT
);
```

- `EmployeeID` starts from `1` and increments by `1` for each new record.

Insert Data

```sql
INSERT INTO Employees (Name, Age) VALUES ('John Doe', 30);
INSERT INTO Employees (Name, Age) VALUES ('Jane Smith', 28);
```

Result

EmployeeID	Name	Age
1	John Doe	30
2	Jane Smith	28

5. Can We Manually Insert Values into an Identity Column?

By default, **you cannot insert a value manually** into an `IDENTITY` column. However, you can **override it** using `SET IDENTITY_INSERT ON`:

```sql
SET IDENTITY_INSERT Employees ON;

INSERT INTO Employees (EmployeeID, Name, Age)
VALUES (100, 'David Brown', 35);

SET IDENTITY_INSERT Employees OFF;
```

---

1. What is the `IDENTITY` property in SQL Server?

Answer:
The `IDENTITY` property in **MS SQL Server** is used to create **auto-incrementing** values for a column, typically used for **primary keys**. It automatically generates unique sequential numbers when a new row is inserted.

Example:

```sql
CREATE TABLE Employees (
    EmployeeID INT IDENTITY(1,1) PRIMARY KEY,
    Name VARCHAR(100)
);
```

Here, `EmployeeID` starts at **1** and increments by **1** for each new row.

2. What are the parameters of `IDENTITY(seed, increment)`?

Answer:

- **Seed** → The starting value of the identity column.
- **Increment** → The value by which the identity column increases.

Example:

```sql
CREATE TABLE Orders (
    OrderID INT IDENTITY(100, 5) PRIMARY KEY
);
```

- **First** `OrderID` = 100
- **Next** `OrderID` = 105, 110, 115, ...

5. What happens if we delete all rows from an identity column?

Answer:

Deleting rows **does not reset** the identity counter.

Example:

```sql
DELETE FROM Employees;
INSERT INTO Employees (Name) VALUES ('Alice');
```

If the last `EmployeeID` was **5**, the next inserted row **will be 6**, **not 1**.

Solution: Reset identity manually:

```sql
DBCC CHECKIDENT ('Employees', RESEED, 0);
```

Now, the next row will start from **1**.