ALTER TABLE table\_name  
ADD column\_name datatype;

ALTER TABLE Customers  
ADD Email varchar(255);

ALTER TABLE table\_name  
DROP COLUMN column\_name;

ALTER TABLE Customers  
DROP COLUMN Email;

ALTER TABLE table\_name  
ALTER COLUMN column\_name datatype;

ALTER TABLE table\_name  
ADD CONSTRAINT constraint\_Name “The constraint”;

ALTER TABLE table\_name  
DROP CONSTRAINT constraint\_name;

ALTER TABLE table\_name  
ADD PRIMARY KEY (ID);

ALTER TABLE table\_name

ADD FOREING KEY ……;

The following **constraints** are commonly used in SQL:

* [NOT NULL](https://www.w3schools.com/sql/sql_notnull.asp) - Ensures that a column cannot have a NULL value
* [UNIQUE](https://www.w3schools.com/sql/sql_unique.asp) - Ensures that all values in a column are different
* [PRIMARY KEY](https://www.w3schools.com/sql/sql_primarykey.asp) - A combination of a NOT NULL and UNIQUE.

Uniquely identifies each row in a table

* [FOREIGN KEY](https://www.w3schools.com/sql/sql_foreignkey.asp) - Prevents actions that would destroy links between tables
* [CHECK](https://www.w3schools.com/sql/sql_check.asp) - Ensures that the values in a column satisfies a specific condition
* [DEFAULT](https://www.w3schools.com/sql/sql_default.asp) - Sets a default value for a column if no value is specified
* [CREATE INDEX](https://www.w3schools.com/sql/sql_create_index.asp) - Used to create and retrieve data from the database very quickly

**NOT NULL**

To create a NOT NULL constraint on the "Age" column when the "Persons" table is already created, use the following SQL:

ALTER TABLE Persons  
Alter column Age int NOT NULL;

**UNIQUE**

To create a UNIQUE constraint on the "ID" column when the table is already created, use the following SQL:

ALTER TABLE Persons  
ADD Constraint UQ\_ID UNIQUE (ID);

ALTER TABLE Persons  
ADD CONSTRAINT UC\_Person UNIQUE (ID,LastName);

ALTER TABLE Persons  
DROP CONSTRAINT UC\_Person;

**PRIMARY KEY**

To allow naming of a PRIMARY KEY constraint, and for defining a PRIMARY KEY constraint on multiple columns, use the following SQL syntax:

CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    CONSTRAINT PK\_Person PRIMARY KEY (ID,LastName)

ALTER TABLE Persons  
ADD PRIMARY KEY (ID);

ALTER TABLE Persons  
ADD CONSTRAINT PK\_Person PRIMARY KEY (ID,LastName);

To drop a PRIMARY KEY constraint, use the following SQL:

ALTER TABLE Persons  
DROP CONSTRAINT PK\_Person;

**FOREIGN KEY**

CREATE TABLE Orders (  
    OrderID int NOT NULL PRIMARY KEY,  
    OrderNumber int NOT NULL,  
    PersonID int FOREIGN KEY REFERENCES Persons(PersonID)  
);

Ou bien

CREATE TABLE Orders (  
    OrderID int NOT NULL,  
    OrderNumber int NOT NULL,  
    PersonID int,  
    PRIMARY KEY (OrderID),  
    CONSTRAINT FK\_PersonOrder FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)  
);

ALTER TABLE Orders  
ADD FOREIGN KEY (PersonID) REFERENCES Persons(PersonID);

ALTER TABLE Orders  
ADD CONSTRAINT FK\_PersonOrder  
FOREIGN KEY (PersonID) REFERENCES Persons(PersonID);

To drop a FOREIGN KEY constraint, use the following SQL:

ALTER TABLE Orders  
DROP FOREIGN KEY FK\_PersonOrder;

**CHECK**

CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int CHECK (Age>=18)  
);

CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    City varchar(255),  
    CONSTRAINT CHK\_Person CHECK (Age>=18 AND Age<=64)  
);

ALTER TABLE Persons  
ADD CHECK (Age>=18);

To allow naming of a CHECK constraint, and for defining a CHECK constraint on multiple columns

ALTER TABLE Persons  
ADD CONSTRAINT CHK\_PersonAge CHECK (Age>=18 AND Age<=64);

To drop a CHECK constraint

ALTER TABLE Persons  
DROP CONSTRAINT CHK\_PersonAge;

**DEFAULT**

CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    City varchar(255) DEFAULT 'Sandnes'  
);

CREATE TABLE Orders (  
    ID int NOT NULL,  
    OrderNumber int NOT NULL,  
    OrderDate date DEFAULT GETDATE()  
);

ALTER TABLE Persons  
ADD CONSTRAINT df\_City DEFAULT 'Sandnes' For city;

ALTER TABLE Persons  
DROP CONSTRAINT df\_City;

**NULL Value**

SELECT column\_namesFROM table\_name  
WHERE column\_name IS NULL;

SELECT column\_namesFROM table\_name  
WHERE column\_name IS NOT NULL;

alter table employee

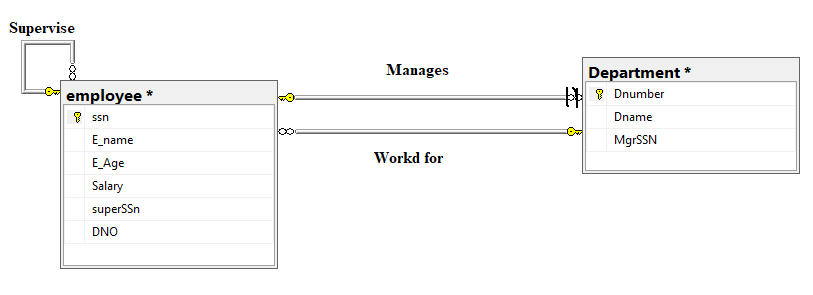
NOCHECK constraint [FK\_\_Employee\_\_DepID\_\_4D5F7D71]

alter table employee

CHECK constraint [FK\_\_Employee\_\_DepID\_\_4D5F7D71]

**Example with constraints**

**version 2**

****

**Business Rules**

1. Un department a u ou plusieurs employees
2. Un employe dirige zero ou un departement
3. Un employe est supervise par un et un seul autre employe
4. Un employe peut superviser zero ou plusieurs employes

create table employee(

ssn char(9) UNIQUE NOT NULL,

E\_name Varchar(100) NOT NULL,

E\_Age Int check(E\_Age>=18),

Salary Decimal(9,2) CHECK (Salary Between(1000 AND 20000),

superSSn char(9) DEFAULT '99999999',

DNO int DEFAULT 1,

PRIMARY Key(ssn),

FOREIGN KEY(superSSn) References Employee(SSN) );

create table Department(

Dnumber int not null,

Dname Varchar(50) NOT NULL,

MgrSSN char(9) NOT NULL DEFAULT '99999999',

Primary key(Dnumber),

Foreign key(MgrSSN) References Employee(SSN) ON DELETE SET DEFAULT ON UPDATE Cascade);

insert into employee values('111111111','Jean',19,12000,'111111111',1);

insert into employee values('222222222','John',27,15000,'111111111',1);

insert into employee values('333333333','George',35,11000,'111111111',2);

insert into employee values('444444444','Jawad',18,13000,'111111111',2);

insert into employee values('555555555','Rami',30,15000,'222222222',2);

insert into employee values('999999999','Rita',19,15000,'999999999',2);

insert into employee (ssn,E\_name,E\_Age,salary,superSSn)values('666666666','Raja',45,12000,'999999999');

insert into department values(1,'Sales',111111111);

insert into department values(2,'HR',111111111);

alter table employee

add constraint FK\_emp\_dept foreign key(DNO) References Department(Dnumber);

Q: Test cascading Foreign key(MgrSSN) References Employee(SSN) ON DELETE SET DEFAULT ON UPDATE Cascade);

Update Employee

set SSn= 111111112

Where SSn='111111111';

select \* from employee;

select \* from department;

**ssn E\_name E\_Age Salary superSSn DNO**

**111111112 Jean 19 12000.00 NULL 1**

**222222222 John 27 15000.00 NULL 1**

**333333333 George 35 11000.00 NULL 2**

**444444444 Jawad 18 13000.00 NULL 2**

**555555555 Rami 30 15000.00 222222222 2**

**666666666 Raja 45 12000.00 999999999 1**

**999999999 Rita 19 15000.00 999999999 2**

**Dnumber Dname MgrSSN**

**1 Sales 111111112**

**2 HR 111111112**

**Date**

**SQL Server** comes with the following data types for storing a date or a date/time value in the database:

* DATE - format YYYY-MM-DD
* DATETIME - format: YYYY-MM-DD HH:MI:SS
* SMALLDATETIME - format: YYYY-MM-DD HH:MI:SS
* TIMESTAMP - format: a unique number

**Note:** The date types are chosen for a column when you create a new table in your database!

SELECT \* FROM Orders WHERE OrderDate='2008-11-11'