# **SQL AUTO INCREMENT Field**

**SQL AUTO INCREMENT Field:**

1. Auto-increment allows a unique number to be generated automatically when a new record is inserted into a table.
2. Often this is the primary key field that we would like to be created automatically every time a new record is inserted.

**Syntax for MySQL**

1. The following SQL statement defines the "Personid" column to be an auto-increment primary key field in the "Persons" table:

|  |
| --- |
| CREATE TABLE Persons (  Personid int NOT NULL AUTO\_INCREMENT,  LastName varchar(255) NOT NULL,  FirstName varchar(255),  Age int,  PRIMARY KEY (Personid)  ); |

1. MySQL uses the AUTO\_INCREMENT keyword to perform an auto-increment feature.
2. By default, the starting value for AUTO\_INCREMENT is 1, and it will increment by 1 for each new record.
3. To let the AUTO\_INCREMENT sequence start with another value, use the following SQL statement:

|  |
| --- |
| ALTER TABLE Persons AUTO\_INCREMENT=100; |

1. To insert a new record into the "Persons" table, we will NOT have to specify a value for the "Personid" column (a unique value will be added automatically):

|  |
| --- |
| INSERT INTO Persons (FirstName,LastName)  VALUES ('Lars','Monsen'); |

1. The SQL statement above would insert a new record into the "Persons" table. The "Personid" column would be assigned a unique value. The "FirstName" column would be set to "Lars" and the "LastName" column would be set to "Monsen".

**Syntax for SQL Server**

1. The following SQL statement defines the "Personid" column to be an auto-increment primary key field in the "Persons" table:

|  |
| --- |
| CREATE TABLE Persons (  Personid int IDENTITY(1,1) PRIMARY KEY,  LastName varchar(255) NOT NULL,  FirstName varchar(255),  Age int  ); |

1. The MS SQL Server uses the IDENTITY keyword to perform an auto-increment feature.
2. In the example above, the starting value for IDENTITY is 1, and it will increment by 1 for each new record.
3. To specify that the "Personid" column should start at value 10 and increment by 5, change it to IDENTITY(10,5).
4. To insert a new record into the "Persons" table, we will NOT have to specify a value for the "Personid" column (a unique value will be added automatically):

|  |
| --- |
| INSERT INTO Persons (FirstName,LastName)  VALUES ('Lars','Monsen'); |

1. The SQL statement above would insert a new record into the "Persons" table. The "Personid" column would be assigned a unique value. The "FirstName" column would be set to "Lars" and the "LastName" column would be set to "Monsen".

**Syntax for Access**

1. The following SQL statement defines the "Personid" column to be an auto-increment primary key field in the "Persons" table:

|  |
| --- |
| CREATE TABLE Persons (  Personid AUTOINCREMENT PRIMARY KEY,  LastName varchar(255) NOT NULL,  FirstName varchar(255),  Age int  ); |

1. The MS Access uses the AUTOINCREMENT keyword to perform an auto-increment feature.
2. By default, the starting value for AUTOINCREMENT is 1, and it will increment by 1 for each new record.
3. To specify that the "Personid" column should start at value 10 and increment by 5, change the autoincrement to AUTOINCREMENT(10,5).
4. To insert a new record into the "Persons" table, we will NOT have to specify a value for the "Personid" column (a unique value will be added automatically):

|  |
| --- |
| INSERT INTO Persons (FirstName,LastName)  VALUES ('Lars','Monsen'); |

1. The SQL statement above would insert a new record into the "Persons" table. The "Personid" column would be assigned a unique value. The "FirstName" column would be set to "Lars" and the "LastName" column would be set to "Monsen".

**Syntax for Oracle**

1. In Oracle the code is a little bit more tricky.
2. You will have to create an auto-increment field with the sequence object (this object generates a number sequence).
3. Use the following CREATE SEQUENCE syntax:

|  |
| --- |
| CREATE SEQUENCE seq\_person  MINVALUE 1  START WITH 1  INCREMENT BY 1  CACHE 10; |

1. The code above creates a sequence object called seq\_person, that starts with 1 and will increment by 1. It will also cache up to 10 values for performance. The cache option specifies how many sequence values will be stored in memory for faster access.
2. To insert a new record into the "Persons" table, we will have to use the nextval function (this function retrieves the next value from seq\_person sequence):

|  |
| --- |
| INSERT INTO Persons (FirstName,LastName)  VALUES ('Lars','Monsen'); |

1. The SQL statement above would insert a new record into the "Persons" table. The "Personid" column would be assigned the next number from the seq\_person sequence. The "FirstName" column would be set to "Lars" and the "LastName" column would be set to "Monsen".