-- Identity : Auto Incremented Value for a column

use employee

create table trainer (trainerid int primary key identity , name varchar(20))

insert into trainer values ('Deepak')

Delete from trainer where trainerid = 2

select \* from trainer

update trainer set trainerid = 6 where name='Deepak'

create table batches (batchid int primary key identity(200,2) , name varchar(20))

insert into batches values ('B002')

select \* from batches

create table b (id int , doj datetime)

SET DATEFORMAT DMY

insert into b values(1,'05/02/2018')

Select \* From b

select CONVERT(VARCHAR(10), doj, 103) from b

--The default Date format in a SQL Server DateTime column is yyyy-MM-dd. SQL Server provided a command named DATEFORMAT which allows us to insert Date in the desired format such as dd/MM/yyyy.

--The DATEFORMAT command has different date format options such as

--1. DMY – dd/MM/yyyy. Ex: 13/06/2018.

--2. YDM – yyyy/dd/MM. Ex: 2018/13/06.

--3. MDY – MM/dd/yyyy. Ex: 06/13/2018.

--4. YMD – yyyy/MM/dd. Ex: 2018/06/13.

--exec sp\_rename oldname, newname renaming table

-- Modularity : Breaking a big program into simple code  -- Normaliztion : Breaking  big table to smaller tables

 --student (id, name , marks, batchid, batchname , marks)

-- To get Records from more than 1 table , we should use Joins -- Foreign Key use Employee

create table student (id int primary key identity , name varchar(20), marks int)

-- will create a column and make it a foreign key  alter table student add batchid int references batch (batchid)

select \* from student create table batch (batchid int primary key , batchname varchar(20))

insert into batch values (1,'B001'), (2,'B002'), (3,'B003') insert into batch values(4,'B004')

select \* from batch

insert into student values('Jay', 89, 1) insert into student values('AJay', 89, 2) insert into student values('Sagar', 89, 1) insert into student values('Deepak', 12, 2)

select \* from student

-- Get list of students with their batch name -- Join these two tables

-- inner join , outer join -- There has to be common column

-- inner join > Matching Records select id,name,marks from student

select student.id, student.name, student.marks,  batch.batchname  from student join batch on student.batchid = batch.batchid

select s.id, s.name, s.marks,  b.batchname  from student s join batch b on s.batchid = b.batchid

-- outer join > Matching as well as Non Matching Records

select s.id, s.name, s.marks,  b.batchname  from student s left outer join batch b on s.batchid = b.batchid

select s.id, s.name, s.marks,  b.batchname  from student s right outer join batch b on s.batchid = b.batchid

select s.id, s.name, s.marks,  b.batchname  from student s full outer join batch b on s.batchid = b.batchid

 -- cross join : This is also known an product of two tables   --no need to have any common column

 drop table batch

create table batch1 (batchid int primary key identity, batchname varchar(20), batchstrength int check(batchstrength between 2 and 10))

create table course (courseid int primary key, coursename varchar(10), duration int check(duration  between 3 and 10))

insert into batch1(batchname, batchstrength) values  ('B001',5), ('B002',10), ('B003',9)

insert into Course(courseid,courseName, duration) values  (1,'C++',3), (2,'Java',8), (3,'Python',9)

select \* from batch1 select \* from course

select batch1.\*, course.\* from batch1 cross join course

select batch1.\*, course.\* into newtable from batch1 cross join course

select \* from newtable