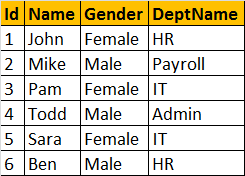
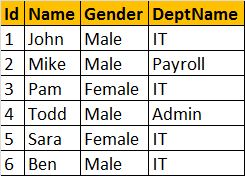
**Is it possible to UPDATE a CTE?**  
**Yes & No**, depending on the number of base tables, the CTE is created upon, and the number of base tables affected by the UPDATE statement. If this is not clear at the moment, don't worry. We will try to understand this with an example.   
  
   
  
   
  
   
  
Let's create the required tblEmployee and tblDepartment tables, that we will be using for this demo.   
  
**SQL Script to create tblEmployee table:**  
CREATE TABLE tblEmployee  
(  
  Id int Primary Key,  
  Name nvarchar(30),  
  Gender nvarchar(10),  
  DepartmentId int  
)  
  
**SQL Script to create tblDepartment table**  
CREATE TABLE tblDepartment  
(  
 DeptId int Primary Key,  
 DeptName nvarchar(20)  
)  
  
**Insert data into tblDepartment table**  
Insert into tblDepartment values (1,'IT')  
Insert into tblDepartment values (2,'Payroll')  
Insert into tblDepartment values (3,'HR')  
Insert into tblDepartment values (4,'Admin')  
  
**Insert data into tblEmployee table**  
Insert into tblEmployee values (1,'John', 'Male', 3)  
Insert into tblEmployee values (2,'Mike', 'Male', 2)  
Insert into tblEmployee values (3,'Pam', 'Female', 1)  
Insert into tblEmployee values (4,'Todd', 'Male', 4)  
Insert into tblEmployee values (5,'Sara', 'Female', 1)  
Insert into tblEmployee values (6,'Ben', 'Male', 3)   
  
**Let's create a simple common table expression**, based on tblEmployee table. **Employees\_Name\_Gender** CTE is getting all the required columns from one base table tblEmployee.  
With Employees\_Name\_Gender  
as  
(  
 Select Id, Name, Gender from tblEmployee  
)  
Select \* from Employees\_Name\_Gender  
  
**Let's now, UPDATE JOHN's gender from Male to Female**, using the **Employees\_Name\_Gender CTE**  
With Employees\_Name\_Gender  
as  
(  
 Select Id, Name, Gender from tblEmployee  
)  
Update Employees\_Name\_Gender Set Gender = 'Female' where Id = 1  
  
**Now, query the tblEmployee table**. JOHN's gender is actually UPDATED. So, if a CTE is created on one base table, then it is possible to UPDATE the CTE, which in turn will update the underlying base table. In this case, UPDATING **Employees\_Name\_Gender**CTE, updates **tblEmployee** table.

**Now, let's create a CTE, on both the tables - tblEmployee and tblDepartment.** The CTE should return, Employee Id, Name, Gender and Department. In short the output should be as shown below.   
   
  
**CTE, that returns Employees by Department**  
With EmployeesByDepartment  
as  
(  
 Select Id, Name, Gender, DeptName   
 from tblEmployee  
 join tblDepartment  
 on tblDepartment.DeptId = tblEmployee.DepartmentId  
)  
Select \* from EmployeesByDepartment  
  
**Let's update this CTE.** Let's change JOHN's Gender from **Female to Male**. Here, the CTE is based on 2 tables, but the UPDATE statement affects only one base table **tblEmployee**. So the UPDATE succeeds. So, if a CTE is based on more than one table, and if the UPDATE affects only one base table, then the UPDATE is allowed.   
With EmployeesByDepartment  
as  
(  
 Select Id, Name, Gender, DeptName   
 from tblEmployee  
 join tblDepartment  
 on tblDepartment.DeptId = tblEmployee.DepartmentId  
)  
Update EmployeesByDepartment set Gender = 'Male' where Id = 1  
  
Now, let's try to **UPDATE the CTE**, in such a way, that the update affects both the tables - **tblEmployee and tblDepartment**. This UPDATE statement changes **Gender**from **tblEmployee** table and **DeptName** from **tblDepartment** table. When you execute this UPDATE, you get an error stating - 'View or function EmployeesByDepartment is not updatable because the modification affects multiple base tables'. So, if a CTE is based on multiple tables, and if the UPDATE statement affects more than 1 base table, then the UPDATE is not allowed.  
With EmployeesByDepartment  
as  
(  
 Select Id, Name, Gender, DeptName   
 from tblEmployee  
 join tblDepartment  
 on tblDepartment.DeptId = tblEmployee.DepartmentId  
)  
Update EmployeesByDepartment set   
Gender = 'Female', DeptName = 'IT'  
where Id = 1  
  
**Finally, let's try to UPDATE just the DeptName**. Let's change JOHN's DeptName from HR to IT. Before, you execute the UPDATE statement, notice that BEN is also currently in HR department.  
With EmployeesByDepartment  
as  
(  
 Select Id, Name, Gender, DeptName   
 from tblEmployee  
 join tblDepartment  
 on tblDepartment.DeptId = tblEmployee.DepartmentId  
)  
Update EmployeesByDepartment set   
DeptName = 'IT' where Id = 1  
  
After you execute the UPDATE. Select data from the CTE, and you will see that BEN's DeptName is also changed to IT.   
   
  
**This is because**, when we updated the **CTE**, the UPDATE has actually changed the **DeptName** from **HR** to **IT**, in **tblDepartment** table, instead of changing the **DepartmentId** column (from 3 to 1) in **tblEmployee** table. So, if a CTE is based on multiple tables, and if the UPDATE statement affects only one base table, the update succeeds. But the update may not work as you expect.  
  
**So in short if,**  
**1.** A CTE is based on a single base table, then the UPDATE suceeds and works as expected.  
**2.** A CTE is based on more than one base table, and if the UPDATE affects multiple base tables, the update is not allowed and the statement terminates with an error.

**3.** A CTE is based on more than one base table, and if the UPDATE affects only one base table, the UPDATE succeeds(but not as expected always)