

## Varendra University Dept. of CSE

**Final Lab Test, Semester: 7<sup>th</sup>**

Create 2 table like “**Teacher**” and “**Department**”. In “**Department**” table ‘DeptId is primary key and “**Teacher**” table “Id” is primary key, “DepartmentID” is foregen key which reference the “**Department**” table.

**Teacher**

ID	FirstName	LastName	Gender	JoiningDate	Salary	DepartmentID
1	Sazzed	Hossain	Male	2016-01-04	30000	1
2	Samia	Sultana	Female	2015-01-01	25000	2
3	Sania	Zahan	Female	2015-06-07	27000	1
4	M.	Rasel	Male	2015-01-09	28000	3
5	Ashriful	Islam	Male	2016-04-05	29000	4
6	Saddam	Hossain	Male	2016-06-06	24000	5
7	Tanzila	Zinat	Female	2012-01-01	2900.5	6

**Department**

DeptId	DeptName	Location	EstablishDate	Founder
1	CSE	Talaimari	2012-12-12	Jon carry
2	EEE	Kajla	2012-12-13	Willem
3	IT	Court	2013-04-06	Rashida Akhter
4	HR	SadurMor	2014-06-11	Saanjir Shishir
5	Payrol	Padma	2015-08-31	Shamim Ahmad
6	Admin	Bihars	2016-04-05	Arifa ferdoshi

1. Create a view like “**vwEmployeeDeptDetails**” in this view find ID, FirstName, LastName, Gender, Salary, DeptName.

Using this view table now create a procedure like “**spTeacherUpdate**”, that have two parameter like @salary, @dept. Parameter type is float, varchar(50). In this procedure update the all teacher’s gender whose Salary is greater than 25000 it change to “Female” and other wise change to “Male” and also check the DeptName is “CSE” where DeptName & Salary pass by parameter. 3

2. Find the ID, DeptName, firstName rank of each Teacher, where ranking measure by “**Joiningdate**”. 2
3. Create a function like “**fnEmployee**” that has two parameter. In this function print the summation of 1-10 values where start value & end value pass by this parameter. 2
4. Using “**vwEmployeeDeptDetails**”. Now In this view insert the data. If you can arise a problem. So write down the query to solve this problem 3

**Varendra University**  
**Dept. of CSE**

**Final Lab Test, Semester: 7<sup>th</sup>**

Create 2 table like “**Teacher**” and “**Department**”. In “**Department**” table ‘DeptId is prymary key and “**Teacher**” table “Id” is primary key, “DepartmentID” is foregen key which reference the “**Department**” table.

**Teacher**

ID	FirstName	LastName	Gender	JoiningDate	Salary	DepartmentID
1	Sazzed	Hossain	Male	2016-01-04	30000	1
2	Samia	Sultana	Female	2015-01-01	25000	2
3	Sania	Zahan	Female	2015-06-07	27000	1
4	M.	Rasel	Male	2015-01-09	28000	3
5	Ashriful	Islam	Male	2016-04-05	29000	4
6	Saddam	Hossain	Male	2016-06-06	24000	5
7	Tanzila	Zinat	Female	2012-01-01	2900.5	6

**Department**

DeptId	DeptName	Location	EstablishDate	Founder
1	CSE	Talaimari	2012-12-12	Jon carry
2	EEE	Kajla	2012-12-13	Willem
3	IT	Court	2013-04-06	Rashida Akhter
4	HR	SadurMor	2014-06-11	Saanjir Shishir
5	Payrol	Padma	2015-08-31	Shamim Ahmad
6	Admin	Bihars	2016-04-05	Arifa ferdoshi

1. Write a SQL Query that Create procedure like “**spAllEnglishCapitalAlphabet**” which has two parameter. In this procedure print all English small letter alphabet where start character & last character value pass by this parameter. 2
2. Find the ID,DeptName, firstName rank of all Employee, where ranking measure by “**Joiningdate**”. 2
3. Create a view like “**vwEmployeeDeptDetails**” in this view find ID, FirstName, LastName, Gender, Salary, DeptName. In this view update the DeptName and Gender. If you can arise a problem. So write down the query to solve this problem. 3
4. Create function like “**fnEmployeeSalaryDetails**” that have four parameter like @salaryStart, @salaryEnd, @gender, @dept. Parameter type is float, float, varchar(50), varchar(50). In this function find the uppercase FirstName & lowercase LastName and added this columes with space and rename the columns “**FULLNAME**”, salary, DeptName, Gender whose Salary between 27000 to 30000 and also check the Gender & Department Name where salary, Gender & Department Name will be pass by parameter. 3

**Varendra University**  
**Dept. of CSE**

**Final Lab Test, Semester: 7<sup>th</sup>**

Create 2 table like “**Teacher**” and “**Department**”. In “**Department**” table ‘DeptId is prymary key and “**Teacher**” table “Id” is primary key, “DepartmentID” is foregen key which reference the “**Department**” table.

**Teacher**

ID	FirstName	LastName	Gender	JoiningDate	Salary	DepartmentID
1	Sazzed	Hossain	Male	2016-01-04	30000	1
2	Samia	Sultana	Female	2015-01-01	25000	2
3	Sania	Zahan	Female	2015-06-07	27000	1
4	M.	Rasel	Male	2015-01-09	28000	3
5	Ashriful	Islam	Male	2016-04-05	29000	4
6	Saddam	Hossain	Male	2016-06-06	24000	5
7	Tanzila	Zinat	Female	2012-01-01	2900.5	6

**Department**

DeptId	DeptName	Location	EstablishDate	Founder
1	CSE	Talaimari	2012-12-12	Jon carry
2	EEE	Kajla	2012-12-13	Willem
3	IT	Court	2013-04-06	Rashida Akhter
4	HR	SadurMor	2014-06-11	Saanjir Shishir
5	Payrol	Padma	2015-08-31	Shamim Ahmad
6	Admin	Bihas	2016-04-05	Arifa ferdoshi

1. Create an index view like “**vwEmployeeSalaryDetails**” in this view find firstName, lastName, number of Salary of each department, average and summation of that salary. 3
2. Create a view like “**vwEmployeeDeptDetails**” in this view find ID, FirstName, LastName, Gender, Salary, DeptName.  
In this view Delete the Teacher whose DeptName is ‘HR’ and ID is 5. If you can arise a problem. So write down the query to solve this problem. 3
3. Create a procedure like “**spSalaryIfo**”. I this procedure find the 3rd largest salary from Teacher. 2
4. Create Inline Function like “**fnEmployee**”, in this function find the Id, FirstName, LastName, DeptName, location where DeptName and location is pass by parameter. 2

**Varendra University**  
**Dept. of CSE**

**Final Lab Test, Semester: 7<sup>th</sup>**

Create 2 table like “**Teacher**” and “**Department**”. In “**Department**” table ‘DeptId is primary key and “**Teacher**” table “Id” is primary key, “DepartmentID” is foregen key which reference the “**Department**” table.

**Teacher**

ID	FirstName	LastName	Gender	JoiningDate	Salary	DepartmentID
1	Sazzed	Hossain	Male	2016-01-04	30000	1
2	Samia	Sultana	Female	2015-01-01	25000	2
3	Sania	Zahan	Female	2015-06-07	27000	1
4	M.	Rasel	Male	2015-01-09	28000	3
5	Ashriful	Islam	Male	2016-04-05	29000	4
6	Saddam	Hossain	Male	2016-06-06	24000	5
7	Tanzila	Zinat	Female	2012-01-01	2900.5	6

**Department**

DeptId	DeptName	Location	EstablishDate	Founder
1	CSE	Talaimari	2012-12-12	Jon carry
2	EEE	Kajla	2012-12-13	Willem
3	IT	Court	2013-04-06	Rashida Akhter
4	HR	SadurMor	2014-06-11	Saanjir Shishir
5	Payrol	Padma	2015-08-31	Shamim Ahmad
6	Admin	Bihars	2016-04-05	Arifa ferdoshi

1. Create a view like “**vwEmployeeDeptDetails**” in this view find ID, FirstName, LastName, Gender, Salary, DeptName.  
Using this view table now create a procedure like “spStudentUpdate”, that have two parameter like @firstName, @dept. Parameter type is float, varchar(50). In this procedure update the all students gender whose firstName length is greater than 5 it change to “Female” and also check the DeptName is “CSE” where DeptName & firstName length value will be pass by parameter. 3
2. Find the ID, DeptName, firstName rank of each Teacher, where ranking measure by “Establishdate”. 2
3. Using Teacher table, Create a trigger name like “tgEmployee”. In this trigger inserted the Teacher and his information all information store another 2 table.(you must be 2 new table) 3
4. Create a function name like VUMAX(@max float) , in this function calculate the max salary and multiply with 1000. Where 1000 pass by parameter. Finally call this function. 2

**II#**

## Set #:

1.

```
create view vwEmployeeDeptDetails
as
select id, firstname,lastname,gender,salary,deptname from employee join Department on
Employee.DepartmentId = Department.DeptId

create proc spTeacherUpdate
@salary float,
@dept varchar(50)
as
begin
update vwEmployeeDeptDetails set Gender=case when salary>@salary then 'Female' else 'Male'
end where deptName=@dept end
```

### Run this procedure:

```
spTeacherUpdate 25000, 'CSE'
```

2.

```
select id, deptname,firstname, rank()over(partition by deptname order by joiningdate) as ranking
from
Employee join Department on Department.DeptId= Employee.DepartmentId
```

3.

```
create function fnEmployee(@start int,@end int )
returns int
As begin
declare @tt int
set @tt=0
while (@start<=@end)
begin
set @tt= @tt + @start
set @start=@start+1

end
return @tt
end
```

### For run this Function:

```
select dbo.fnEmployee(1,10)
```

4.

```
create trigger tr_vwEmployeeDetails
On vwEmployeeDeptDetails
Instead of insert
as
begin
    Declare @DeptId int

    Select @DeptId = DeptId from Department join inserted
    On inserted.DeptName=Department.DeptName
    Insert into tbEmployee (Id,Name,Gender ,DepartmentId)
    select Id,name,gender ,@DeptId from inserted
End
```

## **Set I#:**

**1.**

```
create proc fnEmployee1
@start int,
@end int
As begin
declare @tt int
set @tt=0
while (@start<=@end)
begin

    print char( @start)
    set @start=@start+1
end
end
```

### **For run this procedure:**

```
fnEmployee1 65, 90
```

2.

```
select id, deptname,firstname, rank()over(order by joiningdate) as ranking from Employee join
Department on Department.DeptId= Employee.DepartmentId
```

3.

```
create view vwEmployeeDeptDetails
as
select id, firstname, lastname, gender, salary, deptname from employee join Department on
Employee.DepartmentId = Department.DeptId
```

### Update Solving Query:

```
CREATE trigger tr_EmployeeDetails1
on vwEmployeeDeptDetails
Instead of Update
as
Begin
if(Update(DeptName))
Begin
    Declare @DeptId int

select @DeptId = DeptId from Department join inserted
on inserted.DeptName=Department.DeptName

Update Employee set DepartmentId=@DeptId
from inserted join Employee on Employee.id=inserted.id
end
if(Update(Gender))
begin
    Update tbEmployee set Gender=inserted.Gender
from inserted join tbEmployee on tbEmployee.id=inserted.id
End

End
```

4.

```
create function fnEmployeeSalaryDetails(@a float, @b float, @c varchar(50), @d varchar(50))
returns table
as
return( select (upper(firstname)+ ' ' + lower( lastname)) as fullname , salary, deptname, gender
from Employee join Department on Department.DeptId=Employee.DepartmentId
where salary between @a and @b and Gender=@c and DeptName=@d )
```

### For run this Function:

```
select * from dbo.fnEmployeeSalaryDetails (27000, 30000, 'Male', 'IT')
```

## set :II#

1.

```
create view vwEmployeeDeptDetails
as
select id, firstname,lastname,gender,salary,deptname from employee join Department on
Employee.DepartmentId = Department.DeptId
```

```
create proc spTeacherUpdate
@ firstNamefloat,
@dept varchar(50)
as
begin
update vwEmployeeDeptDetails set Gender=case when len(firstname)>@ firstNamethen
'Female' else 'Male' end where deptName=@dept end
```

### Run this procedure:

```
spTeacherUpdate 5, 'CSE'
```

2.

```
select id, deptname,firstname, rank()over(partition by deptname order by Establishdate) as
ranking from
Employee join Department on Department.DeptId= Employee.DepartmentId
```

3.

```
CREATE TRIGGER tr_Employee_For_Insert
ON tbEmployee
FOR INSERT
AS
BEGIN
    DECLARE @id int
    SELECT @id=Id from inserted
    INSERT INTO tbNewTable1
    VALUES ('New employee with Id =' + Cast(@Id as varchar(5))+
            'is added at ' + cast(Getdate() as varchar(20)) )
    INSERT INTO tbNewTable2
    VALUES ('New employee with Id =' + Cast(@Id as varchar(5))+
            'is added at ' + cast(Getdate() as varchar(20)) )

END
```

4.



```

create Function VUMAX (@Salary float)
returns float
as
begin
declare @maxSalary float
set maxSalary = (select max(@Salary) from Teacher ) * Salary
return @maxSalary
end

```

**Call that function:**

```

select dbo.CalculateSalary (1000)

```

**Set 2##:**

1.

```

create view aa
with schemabinding
as
select id,firstname,lastname,deptname, sum(isnull(salary,0)) as summ,
count_big(*)as ss from dbo.Employee join dbo.Department
on dbo.department.deptid=dbo.employee.departmentid
group by deptname,firstname,lastname,id

```

```

create unique clustered index dd on aa(id)

```

2.

```

Create Trigger tr_vwEmployee_InstedOfD
on vwEmployeeDeptDetails
instead of delete
as
Begin
Delete Employee from Employee join deleted on Employee.Id=deleted.id
End

```

**Run this Query:**

```

delete from vwEmployeeDeptDetails where deptname='HR' and id=5

```

3.

Create proc spSalaryInfo

As begin

```
select max(salary) from Employee where salary<(select max(salary) from Employee where
salary<(select max(salary) from Employee))
end
```

4.

```
create function fnEmployeeSalaryDetails(@dept varchar(50),@location varchar(50))
returns table
as
return( select firstname ,lastname deptname,location from Employee join Department on
Department.DeptId=Employee.DepartmentId
where deptname=@dept and location=@location )
```