

WEEK ONE TASK FOUR

(1)

```
select top 2 salary
from Instructor
order by salary desc
```

(2)

```
select top 2 salary
from(
    select salary, DENSE_RANK() over(order by salary desc)as positionofsalary
    from Instructor
    ) as rankposition
    order by positionofsalary;
```

(3)

```
select StudentID,positionofstudent
from (
    select StudentID, ROW_NUMBER() over(partition by DID order by newid())
    as positionofstudent
    from Student
) as aggregation
where positionofstudent = 1
```

(4)

```
create function getValues (@start int, @end int)
```

```

returns @result table(num int)
as
begin
    declare @value int;
    set @value = @start;
    while @value < @end
        begin
            insert into @result (num)
            values(@value);
            set @value += 1;
        end
    return;
end

```

(5)

```

create function getInformation (@studentID int)
returns @results table(depName varchar(20), fullName varchar(20))
as
begin
    insert into @results(depName, fullName)

        select DName as depName, concat(student.firstName , ' ', student.lastName)
as fullName
        from Student
        where StudentID = @studentID
    return;

```

end

(7)

create function getinformations (@format int)

returns @result table (depatmentName varchar(50), manegerName varchar(50),
hiringDate varchar(50))

as

begin

insert into @result (depatmentName, manegerName, hiringDate)

select DepartmentName as depatmentName,

concat(firstName , ' ' , lastName) as manegerName,

case

when @format = 1 then CONVERT(varchar, hiringDate, 101)

when @format = 2 then CONVERT(varchar, hiringDate, 103)

when @format = 3 then CONVERT(varchar, hiringDate, 104)

else CONVERT(varchar, hiringDate, 120)

end as hiringDate

from Instructor

return;

end

(8)

create function conv (@string varchar(50))

returns @result table (result varchar(50))

as

```
begin
insert into @result(result)
select firName
from Student
where @string = 'first name'
return;
end
```