-- 1 --

select s.salesman\_id,

salesman\_name = s.name,

c.customer\_id,

c.cust\_name,

num\_of\_orders = count(\*)

from salesman s

join customer c on s.salesman\_id = c.salesman\_id

join orders o on c.customer\_id = o.customer\_id

group by s.salesman\_id,

s.name,

c.customer\_id,

c.cust\_name

having count(\*)>1;

-- 2 --

select count(\*)

from customer

where grade >(select avg(grade)

from customer

where city = 'New York' )

-- 3 --

select s.salesman\_id,

salesman\_name = s.name,

num\_of\_cust = count(\*)

from salesman s

join customer c on s.salesman\_id = c.salesman\_id

group by s.salesman\_id,

s.name

having count(\*)>1

-- 4 --

select o.ord\_no,

o.purch\_amt,

o.customer\_id

from orders o join ( select customer\_id,

avg\_ord\_amt = avg(purch\_amt)

from orders

group by customer\_id )a

on o.customer\_id = a.customer\_id

where o.purch\_amt > a.avg\_ord\_amt

go

------------------------------------------------------------------------------------

-- 5 --

if object\_id('usp\_search\_sale','p')is not null

drop procedure usp\_search\_sale

go

create procedure usp\_search\_sale

@xml xml

as

declare

@handle int,

@table sysname = 'Salesman', -- future parameter

@column sysname,

@cllist varchar(max),

@sql nvarchar(max)

set nocount on

exec sp\_xml\_preparedocument @handle output, @xml

select col

into #temp

from openxml (@handle, '/table/columns/column',2)

with (col varchar(20) '.')

exec sp\_xml\_removedocument @handle

set @cllist =''

declare cr cursor

for

select col

from #temp

open cr

fetch cr into @column

while @@fetch\_status = 0

begin

set @cllist += @column + ','

fetch cr into @column

end

close cr

deallocate cr

set @cllist = left(@cllist, len(@cllist)-1)

set @sql = 'select ' + @cllist + ' from ' + @table

--print @sql

set nocount off

begin try

exec sp\_executesql @sql

end try

begin catch

declare

@error\_number int,

@error\_message varchar(max),

@error\_line int,

@msg varchar(max),

@error\_proc varchar(256)

select @error\_number = error\_number(),

@error\_message= error\_message(),

@error\_line = error\_line(),

@error\_proc = error\_procedure()

set @msg = 'An error number ' + cast(@error\_number as varchar(6)) + ': ' + @error\_message + ' occured on line ' + cast(@error\_line as varchar(6)) + ' in ' + isnull(@error\_proc,'') + ' procedure. '

raiserror (@msg, 16,1)

return

end catch

go

execute usp\_search\_sale

@xml =

'

<table>

<columns>

<column>name</column>

<column>city</column>

</columns>

</table>

'