

=====Simulado Prova de Subconsultas e procedures=====

2-Faça uma Subconsulta que mostre o **nome da construtora** e a **quantidade total de estradas** que cada empresa construiu. Mostre ordenado, do **maior para menor**, pelo total construído (**soma do comprimento**).

```
SELECT COUNT(nomeConstrutora) AS
'quantidadeTotalEstrada', nomeConstrutora
FROM trecho
WHERE nomeConstrutora IN(
    SELECT nomeConstrutora
    FROM trecho
)
GROUP BY nomeConstrutora
ORDER BY SUM(comprimento) DESC
```

3 – Faça uma subconsulta para mostrar o **nome das localidades** e a **quantidade de trechos** que passam por cada uma dessas localidades.

```
SELECT l.NomeLoc, COUNT(t.sequencia ) AS 'quantidadeTrecho',
FROM localidade AS l
INNER JOIN trecho AS t
ON l.codLoc = t.codLocIn
GROUP BY l.nomeLoc
```

4 – Faça uma Stored Procedure que receba **com parâmetro o nome da unidade de federação Inicial e da unidade de federação final**. Com estas informações, mostre os nomes de **todas estradas** que têm pelo menos um trecho que começa na unidade da federação informada no parâmetro “**federação inicial**” e que termina na unidade da **federação informada** no último parâmetro. Exemplo: A procedure é executada passando os parâmetros ‘RJ’ e ‘MG’, em seguida é retornado o nome de todas as estradas que começam em ‘RJ’ e terminam em ‘MG’.

```

CREATE PROCEDURE VerificarFederacao (@federacaoInicial
varchar(2),@federacaoFinal varchar(2))
AS
BEGIN
    SELECT e.NomeEst
    FROM estrada AS e
    INNER JOIN Trecho AS t
    ON e.CodEst = t.codEst
    INNER JOIN localidade as l
    ON t.codLocIni = l.codLoc
    INNER JOIN UF AS u
    ON l.SiglaUF = u.SiglaUF
    WHERE u.SiglaUF = @federacaoInicial AND u.SiglaUF =
@federacaoFinal

END;

```

=====

5 – Faça uma Stored Procedure para exibir os **nomes das estradas** que têm trechos **construídos por uma construtora** informada no parâmetro desta função. Observar que a coluna Trecho. Nome Construtora é opcional.

```

CREATE PROCEDURE VerificarNomeConstrutora (@nomeConstrutora
Varchar(50))
AS
BEGIN
    SELECT e.nomeEst, t.NomeConstrutora
    FROM Estrada AS e
    INNER JOIN trecho as t
    ON e.CodEst = t.CodEst
    WHERE NomeConstrutora = @nomeConstrutora

END;

```

=====

QUESTÃO QUE EU ME ORGULHEI DE FORMULAR E CONSEGUI RESOLVER

1)Quero poder retornar a quantidade total de itens comprado por cada cliente,nesse sentido quero saber qual o nome do vendedor que participou dessa soma total por pedido e quem foi o cliente que fez cada pedido

```
SELECT resultado.pedido, SUM(resultado.quantidade) AS
'TotalPedido',resultado.nome_vendedor,resultado.nome_cliente
FROM
(SELECT it.num_pedido as 'pedido',it.quantidade
,c.Nome_cliente,v.nome_vendedor
FROM dbo.item_pedido as it
INNER JOIN dbo.pedido as p
ON it.num_pedido = p.num_pedido
INNER JOIN dbo.cliente as c
ON p.cod_cliente = c.cod_cliente
INNER JOIN vendedor as v
ON p.cod_vendedor = v.cod_vendedor
) AS resultado
GROUP BY resultado.pedido,
resultado.nome_vendedor,resultado.nome_cliente;
```

=====

RESULÇÃO da QUESTÃO 3 ALTERNATIVA

```
SELECT SUM(resultado.sequencia) AS
'quantidadeTrecho',resultado.nomeLoc
FROM (
    SELECT t.sequencia, l.nomeLoc
    FROM trecho as t
    INNER JOIN localidade as l
    ON t.codLocIni = l.codLoc
) AS resultado
GROUP BY resultado.nomeLoc
```

OBS: quando envolver uma tabela juntando com outra preferir esse método