

## Mão na massa: Relatórios

Chegou a hora de você executar o que foi visto na aula! Para isso, baixe [aqui](https://s3.amazonaws.com/caelum-online-public/836-consultas-com-sql-server-2017/06/Downloads+-+Aula+6.zip) (<https://s3.amazonaws.com/caelum-online-public/836-consultas-com-sql-server-2017/06/Downloads+-+Aula+6.zip>) os arquivos necessários e siga os passos abaixo.

### Relatório de vendas válidas

1) Objetivo desta prática é a de montar o relatório de vendas por cliente e comparar se o total de venda (quantidade) em um determinado mês superou ou não o limite máximo de venda por cliente (quantidade limite contida no cadastro):

	NOME	MES_ANO	QUANTIDADE	(Nenhum nome de coluna)
1	Abel Silva	2015-01	23176	VENDA VÁLIDA
2	Abel Silva	2015-02	21325	VENDA VÁLIDA
3	Abel Silva	2015-03	23925	VENDA VÁLIDA
4	Abel Silva	2015-04	22390	VENDA VÁLIDA
5	Abel Silva	2015-05	22567	VENDA VÁLIDA
6	Abel Silva	2015-06	22851	VENDA VÁLIDA
7	Abel Silva	2015-07	19573	VENDA VÁLIDA
8	Abel Silva	2015-08	20160	VENDA VÁLIDA
9	Abel Silva	2015-09	22153	VENDA VÁLIDA
10	Abel Silva	2015-10	22114	VENDA VÁLIDA
11	Abel Silva	2015-11	19077	VENDA VÁLIDA
12	Abel Silva	2015-12	21968	VENDA VÁLIDA
13	Abel Silva	2016-01	26541	VENDA INVÁLIDA
14	Abel Silva	2016-02	21912	VENDA VÁLIDA
15	Abel Silva	2016-03	22377	VENDA VÁLIDA
16	Abel Silva	2016-04	20908	VENDA VÁLIDA
17	Abel Silva	2016-05	24125	VENDA VÁLIDA
18	Abel Silva	2016-06	19353	VENDA VÁLIDA

2) Abra o arquivo **Vendas válidas EXEMPLO.sql**.

3) Há 3 tabelas que você irá usar para montar este relatório. A tabela de clientes, para buscar os dados do clientes (nome, por exemplo) e também o valor total de crédito contido no seu cadastro. A outra tabela é a de notas fiscais, onde tem a data da venda. A ligação entre estas duas tabelas é o CPF do cliente. Finalmente, você irá acessar a tabela de itens de nota fiscais, onde você obtém o valor faturado. Esta última tabela irá se juntar com a tabela de notas fiscais pelo número da nota:

```
SELECT * FROM [TABELA DE CLIENTES]
SELECT * FROM [ITENS NOTAS FISCAIS]
SELECT * FROM [NOTAS FISCAIS]
```

4) Primeiramente, junte a tabela de notas fiscais com a tabela de itens:

```
SELECT * FROM [NOTAS FISCAIS] NF
INNER JOIN [ITENS NOTAS FISCAIS] INF
ON NF.NUMERO = INF.NUMERO
```

	CPF	MATRICULA	DATA	NUMERO	IMPOSTO	NUMERO	CÓDIGO DO PRODUTO	QUANTIDADE	PREÇO
1	7771579779	00235	2015-01-01	100	0,1	100	1013793	63	24,01
2	7771579779	00235	2015-01-01	100	0,1	100	1101035	26	9,0105
3	7771579779	00235	2015-01-01	100	0,1	100	520380	67	12,0111
4	7771579779	00235	2015-01-01	100	0,1	100	773912	66	8,008
5	50534475787	00237	2015-01-01	101	0,12	101	1000689	35	6,309
6	50534475787	00237	2015-01-01	101	0,12	101	479745	65	3,768
7	8502682733	00236	2015-01-01	102	0,12	102	1101035	84	9,0105
8	8502682733	00236	2015-01-01	102	0,12	102	235653	37	3,8595
9	8502682733	00236	2015-01-01	102	0,12	102	394479	74	8,409
10	5840119709	00235	2015-01-01	103	0,12	103	520380	29	12,0111

5) A seguir, segregue os campos importantes deste `JOIN`, aplicando algumas funções. Você irá obter apenas o CPF, a data convertida para mês e ano (para ver somente os dados dentro do mês) e a quantidade vendida:

```
SELECT NF.CPF, SUBSTRING(CONVERT(VARCHAR, NF.[DATA], 120), 1, 7) AS ANO_MES,
       INF.QUANTIDADE FROM [NOTAS FISCAIS] NF
  INNER JOIN [ITENS NOTAS FISCAIS] INF
  ON NF.NUMERO = INF.NUMERO
```

	CPF	DATA_CONVERTIDA	QUANTIDADE
361	50534475787	2015-01-03	65
362	564641702	2015-01-03	91
363	5576228758	2015-01-03	79
364	5576228758	2015-01-03	19
365	5576228758	2015-01-03	21
366	8719655770	2015-01-03	17
367	8719655770	2015-01-03	91
368	1471156710	2015-01-03	40
369	1471156710	2015-01-03	78
370	1471156710	2015-01-03	21

6) Agrupe esta consulta para obter o valor total, por cliente e por mês/ano:

```
SELECT NF.CPF, SUBSTRING(CONVERT(VARCHAR, NF.[DATA], 120), 1, 7) AS ANO_MES,
       SUM(INF.QUANTIDADE) AS QUANTIDADE_MES FROM [NOTAS FISCAIS] NF
  INNER JOIN [ITENS NOTAS FISCAIS] INF
  ON NF.NUMERO = INF.NUMERO
 GROUP BY NF.CPF, SUBSTRING(CONVERT(VARCHAR, NF.[DATA], 120), 1, 7)
```

	CPF	ANO_MES	QUANTIDADE_MES
1	9283760794	2016-04	23352
2	9283760794	2017-07	21705
3	8719655770	2016-09	22665
4	9283760794	2015-02	19614
5	8502682733	2016-06	23751
6	1471156710	2017-08	20377
7	2600586709	2015-07	21672
8	8502682733	2017-04	19405
9	564641702	2016-10	21866
10	19290992743	2017-06	20034

7) Guarde esta consulta e verifique, agora, os dados do limite de vendas em quantidades contidas na tabela de clientes. Selecione apenas os campos necessários a serem usados no relatório:

```
SELECT TC.NOME, TC.[VOLUME DE COMPRA] FROM [TABELA DE CLIENTES] TC
```

	NOME	VOLUME DE COMPRA
1	Erica Carvalho	24500
2	Fernando Cavalcante	20000
3	César Teixeira	22000
4	Marcos Nougueira	22000
5	Eduardo Jorge	9500
6	Abel Silva	26000
7	Petra Oliveira	16000
8	Paulo César Mattos	22000
9	Gabriel Araujo	21000
10	Marcelo Mattos	20000

8) Junte as duas consultas acima. A que obtém dados de vendas e a que obtém dados do limite máximo:

```
SELECT TC.NOME, CQ.ANO_MES, CQ.QUANTIDADE_MES, TC.[VOLUME DE COMPRA]
FROM
(SELECT NF.CPF, SUBSTRING(CONVERT(VARCHAR, NF.[DATA], 120),1,7) AS ANO_MES,
SUM(INF.QUANTIDADE) AS QUANTIDADE_MES FROM [NOTAS FISCAIS] NF
INNER JOIN [ITENS NOTAS FISCAIS] INF
ON NF.NUMERO = INF.NUMERO
GROUP BY NF.CPF, SUBSTRING(CONVERT(VARCHAR, NF.[DATA], 120),1,7)) CQ
INNER JOIN [TABELA DE CLIENTES] TC ON TC.CPF = CQ.CPF
```

	NOME	ANO_MES	QUANTIDADE_MES	VOLUME DE COMPRA
1	Edson Meilleles	2015-04	23352	25000
2	Edson Meilleles	2017-07	1705	25000
3	Carlos Eduardo	2016-09	22665	24000
4	Edson Meilleles	2015-02	19614	25000
5	Valdeci da Silva	2016-06	23751	19000
6	Erica Carvalho	2017-08	20377	24500
7	César Teixeira	2015-07	21672	22000
8	Valdeci da Silva	2017-04	19405	19000
9	Paulo César Mattos	2016-10	21866	22000
10	Fernando Cavalcante	2017-06	20034	20000

9) Com duas colunas comparando as vendas com os limites, crie um `CASE` que irá testar se a venda do mês superou ou não a venda limite e escrever um *label* para isso. Apresente esta consulta ordenada e você terá o relatório final:

```
SELECT AUX1.NOME, AUX1.ANO_MES, AUX1.QUANTIDADE_MES,
CASE WHEN AUX1.QUANTIDADE_MES <= AUX1.[VOLUME DE COMPRA] THEN 'VENDA VÁLIDA'
WHEN AUX1.QUANTIDADE_MES > AUX1.[VOLUME DE COMPRA] THEN 'VENDA INVÁLIDA'
END AS STATUS_VENDA
FROM
(SELECT TC.NOME, CQ.ANO_MES, CQ.QUANTIDADE_MES, TC.[VOLUME DE COMPRA]
FROM
(SELECT NF.CPF, SUBSTRING(CONVERT(VARCHAR, NF.[DATA], 120),1,7) AS ANO_MES,
SUM(INF.QUANTIDADE) AS QUANTIDADE_MES FROM [NOTAS FISCAIS] NF
INNER JOIN [ITENS NOTAS FISCAIS] INF
ON NF.NUMERO = INF.NUMERO
GROUP BY NF.CPF, SUBSTRING(CONVERT(VARCHAR, NF.[DATA], 120),1,7)) CQ
INNER JOIN [TABELA DE CLIENTES] TC ON TC.CPF = CQ.CPF) AUX1
ORDER BY AUX1.NOME, AUX1.ANO_MES
```

	NOME	ANO_MES	QUANTIDADE_MES	STATUS_VENDA
1	Abel Silva	2015-01	23176	VENDA VÁLIDA
2	Abel Silva	2015-02	21325	VENDA VÁLIDA
3	Abel Silva	2015-03	23925	VENDA VÁLIDA
4	Abel Silva	2015-04	22390	VENDA VÁLIDA
5	Abel Silva	2015-05	22567	VENDA VÁLIDA
6	Abel Silva	2015-06	22851	VENDA VÁLIDA
7	Abel Silva	2015-07	19573	VENDA VÁLIDA
8	Abel Silva	2015-08	20160	VENDA VÁLIDA
9	Abel Silva	2015-09	22153	VENDA VÁLIDA
10	Abel Silva	2015-10	22192	VENDA VÁLIDA
11	Abel Silva	2015-11	19077	VENDA VÁLIDA
12	Abel Silva	2015-12	21968	VENDA VÁLIDA
13	Abel Silva	2016-01	26541	VENDA INVÁLIDA
14	Abel Silva	2016-02	21912	VENDA VÁLIDA
15	Abel Silva	2016-03	22377	VENDA VÁLIDA
16	Abel Silva	2016-04	20908	VENDA VÁLIDA
17	Abel Silva	2016-05	24125	VENDA VÁLIDA
18	Abel Silva	2016-06	19353	VENDA VÁLIDA

Consulta executada com êxito. | DESKTOP-N51LB16 (14.0 RTM) | sa (54) | SUCOS\_VENDAS | 00:00:00 | 546 linhas

## Relatório de vendas por sabor

10) O objetivo é montar um relatório de vendas (faturamento) por sabor, dentro de um período de tempo (no caso, no ano de 2016) e criar uma coluna com o percentual de participação de cada venda. É importante exibir os dados ordenados por valor do faturamento, do maior para o menor:

	SABOR	ANO	FATURAMENTO	PARTICIPACAO
1	Açaí	2016	10024014.10	23.6627%
2	Melância	2016	6128144.29	14.4661%
3	Manga	2016	5725142.63	13.5148%
4	Maçã	2016	4321369.74	10.201%
5	Cereja/...	2016	4071635.98	9.6115%
6	Laranja	2016	3844934.55	9.07635%
7	Maracujá	2016	1987886.75	4.6926%
8	Morango	2016	1913604.25	4.51725%
9	Lima/Li...	2016	1508300.36	3.56049%
10	Cereja	2016	1063754.48	2.5111%
11	Morango	2016	974442.27	2.30027%
12	Uva	2016	798888.80	1.88586%

11) Abra o arquivo **Vendas por sabor EXEMPLO.sql**.

12) Esta consulta envolverá a tabela de produtos, para buscar os sabores, a tabela de itens de notas fiscais, para obter o faturamento (quantidade X preço), e a tabela de notas fiscais, onde tem a data da venda. Você irá juntar a tabela de produtos e a tabela de itens de notas fiscais pelo código de produto, e em seguida juntará com a tabela de notas fiscais:

```
SELECT * FROM [TABELA DE PRODUTOS] TP
SELECT * FROM [NOTAS FISCAIS] NF
SELECT * FROM [ITENS NOTAS FISCAIS] INF
```

13) Junte as três tabelas, selecionando os campos que são importantes para o relatório:

```
SELECT TP.SABOR, NF.DATA, (INF.QUANTIDADE * INF.PREÇO) AS FATURAMENTO
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CÓDIGO DO PRODUTO] = INF.[CÓDIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
```

	SABOR	DATA	FATURAMENTO
1	Cereja/Maçã	2015-01-01	1512,63
2	Morango/Limão	2015-01-01	234,273
3	Maçã	2015-01-01	804,737
4	Laranja	2015-01-01	528,528
5	Uva	2015-01-01	220,815
6	Laranja	2015-01-01	244,92
7	Morango/Limão	2015-01-01	756,882
8	Manga	2015-01-01	142,8015

Consulta executada com êxito. DESKTOP-N51LB16 (14.0 RTM) sa (54) SUCOS\_VENDAS 00:00:00 213364 linhas

14) Agrupe esta consulta por sabor e ano:

```
SELECT TP.SABOR, YEAR(NF.DATA) AS ANO, SUM (INF.QUANTIDADE * INF.PREÇO) AS FATURAMENTO
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CODIGO DO PRODUTO] = INF.[CODIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
GROUP BY TP.SABOR, YEAR(NF.DATA)
```

	SABOR	ANO	FATURAMENTO
1	Açaí	2015	9345514,908
2	Cereja	2015	1028420,7
3	Cereja/Maçã	2015	3953306,06
4	Laranja	2015	3653928,152
5	Lima/Limão	2015	1386509,452
6	Maçã	2015	4135919,3070001
7	Manga	2015	5321977,0015
8	Maracujá	2015	1794187,536

Consulta executada com êxito. DESKTOP-N51LB16 (14.0 RTM) sa (54) SUCOS\_VENDAS 00:00:00 48 linhas

15) Aplique o filtro para ter somente as vendas de 2016:

```
SELECT TP.SABOR, YEAR(NF.DATA) AS ANO, SUM (INF.QUANTIDADE * INF.PREÇO) AS FATURAMENTO
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CODIGO DO PRODUTO] = INF.[CODIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
WHERE YEAR(NF.DATA) = 2016
GROUP BY TP.SABOR, YEAR(NF.DATA)
```

	SABOR	ANO	FATURAMENTO
6	Maçã	2016	4321369,73729999
7	Manga	2016	5725142,62582499
8	Maracujá	2016	1987886,74560001
9	Melância	2016	6128144,29499999
10	Morango	2016	974442,268800002
11	Morango/Li...	2016	1913604,25095
12	Uva	2016	798888,796650002

16) Guarde esta consulta e faça outra para obter a soma total de vendas no mesmo período. Para isso, aproveite a consulta anterior, mas tire o sabor do grupo:

```
SELECT YEAR(NF.DATA) AS ANO, SUM (INF.QUANTIDADE * INF.PREÇO) AS FATURAMENTO
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CODIGO DO PRODUTO] = INF.[CODIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
WHERE YEAR(NF.DATA) = 2016
```

GROUP BY YEAR(NF.DATA)

Resultados		Mensagens
ANO	FATURAMENTO	
2016	42362118,1877254	

17) Junte as duas consultas pelo campo que representa o ano:

```

SELECT AUX1.SABOR, AUX1.ANO, AUX1.FATURAMENTO, AUX2.TOTAL FROM
(SELECT TP.SABOR, YEAR(NF.DATA) AS ANO, SUM (INF.QUANTIDADE * INF.PREÇO) AS FATURAMENTO
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CODIGO DO PRODUTO] = INF.[CODIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
WHERE YEAR(NF.DATA) = 2016
GROUP BY TP.SABOR, YEAR(NF.DATA)) AUX1
INNER JOIN (SELECT YEAR(NF.DATA) AS ANO, SUM (INF.QUANTIDADE * INF.PREÇO) AS TOTAL
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CODIGO DO PRODUTO] = INF.[CODIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
WHERE YEAR(NF.DATA) = 2016
GROUP BY YEAR(NF.DATA)) AUX2
ON AUX1.ANO = AUX2.ANO

```

SABOR	ANO	FATURAMENTO	TOTAL
Açaí	2016	10024014.105	42362118,1877254
Cereja	2016	1063754,47709999	42362118,1877254
Cereja/Maçã	2016	4071635,9775	42362118,1877254
Laranja	2016	3844934,5452	42362118,1877254
Lima/Limão	2016	1508300,3628	42362118,1877254
Maçã	2016	4321369,73729999	42362118,1877254
Manga	2016	5725142,62582499	42362118,1877254
Maracujá	2016	1987886,74560001	42362118,1877254

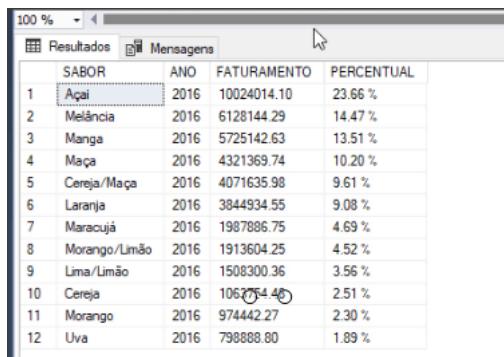
18) Para encerrar o relatório, calcule a coluna percentual e ordene a saída do maior para o menor faturamento. Aproveite e converta os dados de faturamento e percentual para ser representado com duas casas decimais:

```

SELECT AUX1.SABOR, AUX1.ANO, CONVERT(DECIMAL(15,2), AUX1.FATURAMENTO) AS FATURAMENTO,
       CONVERT(VARCHAR, CONVERT(DECIMAL(15,2),(AUX1.FATURAMENTO/AUX2.TOTAL) * 100)) + ' %'
AS PERCENTUAL FROM
(SELECT TP.SABOR, YEAR(NF.DATA) AS ANO, SUM (INF.QUANTIDADE * INF.PREÇO) AS FATURAMENTO
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CODIGO DO PRODUTO] = INF.[CODIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
WHERE YEAR(NF.DATA) = 2016
GROUP BY TP.SABOR, YEAR(NF.DATA)) AUX1
INNER JOIN (SELECT YEAR(NF.DATA) AS ANO, SUM (INF.QUANTIDADE * INF.PREÇO) AS TOTAL
FROM [ITENS NOTAS FISCAIS] INF INNER JOIN [TABELA DE PRODUTOS] TP
ON TP.[CODIGO DO PRODUTO] = INF.[CODIGO DO PRODUTO]
INNER JOIN [NOTAS FISCAIS] NF
ON NF.NUMERO = INF.NUMERO
WHERE YEAR(NF.DATA) = 2016
GROUP BY YEAR(NF.DATA)) AUX2

```

```
ON AUX1.ANO = AUX2.ANO
ORDER BY AUX1.FATURAMENTO DESC
```



	SABOR	ANO	FATURAMENTO	PERCENTUAL
1	Açaí	2016	10024014.10	23.66 %
2	Melância	2016	6128144.29	14.47 %
3	Manga	2016	5725142.63	13.51 %
4	Maçã	2016	4321369.74	10.20 %
5	Cereja/Maçã	2016	4071635.98	9.61 %
6	Laranja	2016	3844934.55	9.08 %
7	Maracujá	2016	1987886.75	4.69 %
8	Morango/Limão	2016	1913604.25	4.52 %
9	Lima/Limão	2016	1508300.36	3.56 %
10	Cereja	2016	1062754.46	2.51 %
11	Morango	2016	974442.27	2.30 %
12	Uva	2016	798888.80	1.89 %