

# Funções de agregação (count, sum, max, min, avg, median, first, last, partition by)

Confira algumas funções que ajudam nas consultas, onde é necessário criar agrupamentos de diferentes formas.

## TABELA EXEMPLO

### EX\_ITENS

CODIGO	DESCRICAO	TIPO	QUANTIDADE
1	Biscoito	Comida	20
2	Água	Bebida	47,95
3	Suco	Bebida	23,4
4	Pão	Comida	10
5	Refrigerante	Bebida	13,75

### SQL de CRIAÇÃO/INSERÇÃO:

```
CREATE TABLE ex_itens (  
  codigo INTEGER,  
  descricao VARCHAR2(200),  
  tipo VARCHAR(20),  
  quantidade DECIMAL(15,2));  
  
INSERT INTO ex_itens VALUES (1, 'Biscoito', 'Comida', 20);  
INSERT INTO ex_itens VALUES (2, 'Água', 'Bebida', 47.95);  
INSERT INTO ex_itens VALUES (3, 'Suco', 'Bebida', 23.4);  
INSERT INTO ex_itens VALUES (4, 'Pão', 'Comida', 10);  
INSERT INTO ex_itens VALUES (5, 'Refrigerante', 'Bebida', 13.75);
```

```
COMMIT;
```

## Consulta com agrupamento agregado

```
SELECT COUNT( QUANTIDADE) QTD_POR_TIPO,
       SUM( QUANTIDADE) VOLUME_POR_TIPO,
       MAX( DESCRICAO) KEEP ( DENSE_RANK FIRST ORDER BY QUANTIDADE)
ITEM_MENOR_QTD_POR_TIPO,
       MIN( QUANTIDADE) MENOR_QTD_POR_TIPO,
       MAX( DESCRICAO) KEEP ( DENSE_RANK LAST ORDER BY QUANTIDADE)
ITEM_MAIOR_QTD_POR_TIPO,
       MAX( QUANTIDADE) MAIOR_QTD_POR_TIPO,
       TRUNC( AVG( QUANTIDADE), 2) MEDIA_QTD_POR_TIPO,
       MEDIAN( QUANTIDADE) MEDIANA_QTD_POR_TIPO
FROM EX_ITENS
GROUP BY TIPO
```

### Retorno:

QTD_POR_TIPO	VOLUME_POR_TIPO	ITEM_MENOR_QTD_POR_TIPO	MENOR_QTD_POR_TIPO	ITEM_MAIOR_QTD_POR_TIPO	MAIOR_QTD_POR_TIPO	MEDIA_QTD_POR_TIPO	MEDIANA_QTD_POR_TIPO
3	85.1	' Refrigerante'	13.75	' Água'	47.95	28.36	23.4
2	30	' Pão'	10	' Biscoito'	20	15	15

- **Count:** Retorna quantidade de linhas no agrupamento definido;
- **Sum:** Retorna soma de uma coluna das linhas no agrupamento definido;
- **Max:** Retorna valor máximo do campo de entrada no agrupamento definido;
- **Min:** Retorna valor mínimo do campo de entrada no agrupamento definido;
- **First:** Retorna primeiro resultado pela ordenação definida, utilizando estrutura com **Keep** e **Dense\_rank**;
- **Last:** Retorna último resultado pela ordenação definida, utilizando estrutura com **Keep** e **Dense\_rank**;
- **Avg:** Retorna média dos valores do campo de entrada no agrupamento definido;
- **Median:** Retorna mediana dos valores do campo de entrada no agrupamento definido.

## Consulta com agrupamento analítico

```
SELECT CODIGO, DESCRICAO, TIPO, QUANTIDADE,
```

```

DENSE_RANK() OVER (ORDER BY QUANTIDADE DESC) RANK_QTD,
DENSE_RANK() OVER (PARTITION BY TIPO ORDER BY QUANTIDADE DESC)
RANK_QTD_POR_TIPO,
COUNT(QUANTIDADE) OVER (PARTITION BY TIPO) QTD_ITENS_POR_TIPO,
SUM(QUANTIDADE) OVER (PARTITION BY TIPO) QTD_VOLUME_POR_TIPO,
MAX(DESCRICAO) KEEP (DENSE_RANK FIRST ORDER BY QUANTIDADE) OVER (PARTITION BY TIPO) AS
ITEM_MENOR_QTD_POR_TIPO,
MIN(QUANTIDADE) OVER (PARTITION BY TIPO) MENOR_QTD_POR_TIPO,
MAX(DESCRICAO) KEEP (DENSE_RANK LAST ORDER BY QUANTIDADE) OVER (PARTITION BY TIPO) AS
ITEM_MAIOR_QTD_POR_TIPO,
MAX(QUANTIDADE) OVER (PARTITION BY TIPO) MAIOR_QTD_POR_TIPO,
TRUNC(AVG(QUANTIDADE) OVER (PARTITION BY TIPO), 2) MEDIA_QTD_POR_TIPO,
MEDIAN(QUANTIDADE) OVER (PARTITION BY TIPO) MEDIANA_QTD_POR_TIPO
FROM EX_ITENS

```

## Retorno:

CODIGO	DESCRICAO	TIPO	QUANTIDADE	RANK_QTD	RANK_QTD_POR_TIPO	QTD_ITENS_POR_TIPO	QTD_VOLUME_POR_TIPO	ITEM_MENOR_QTD_POR_TIPO	MENOR_QTD_POR_TIPO	ITEM_MAIOR_QTD_POR_TIPO	MAIOR_QTD_POR_TIPO	MEDIA_QTD_POR_TIPO	MEDIANA_QTD_POR_TIPO
2	'Água'	'Bebida'	47.95	1	1	3	85.1	'Refrigerante'	13.75	'Água'	47.95	28.36	23.4
3	'Suco'	'Bebida'	23.4	2	2	3	85.1	'Refrigerante'	13.75	'Água'	47.95	28.36	23.4
1	'Biscoito'	'Comida'	20	3	1	2	30	'Pão'	10	'Biscoito'	20	15	15
5	'Refrigerante'	'Bebida'	13.75	4	3	3	85.1	'Refrigerante'	13.75	'Água'	47.95	28.36	23.4
4	'Pão'	'Comida'	10	5	2	2	30	'Pão'	10	'Biscoito'	20	15	15

- **Dense\_rank():** Retorna posição de rank definida pela ordenação na estrutura com **Over**;
- **Partition by:** Utilizado para retornar um valor agrupado sem fazer um **distinct**.

Revisão #2

Criado 9 March 2022 14:52:37 por Nicolly Andrielly

Atualizado 10 March 2022 14:23:40 por Nicolly Andrielly