



Aula 20

Prof: Henrique Augusto Maltauro

Implementar Banco de Dados Para WEB

SQL

- DQL: SELECT

O SQL possui algumas **funções internas**, da mesma maneira que as **linguagens de programação**, que irão realizar algum processo já pré determinado.

E elas são executadas da mesma maneira que nós as executamos em uma **linguagem de programação**, **pelo nome delas e fazendo o uso do parênteses para os parâmetros.**

SQL

- **DQL: SELECT (COALESCE)**

O COALESCE é uma **função** do **SQL**, que recebe dois **parâmetros**.

O **primeiro parâmetro** vai ser uma coluna, que será apresentada na consulta.

O **segundo parâmetro** vai ser alguma informação que será apresentada **SOMENTE** se a coluna do primeiro parâmetro possuir um valor nulo.

SQL

- DQL: SELECT (COALESCE)

```
SELECT  
    P.ID,  
    P.NOME,  
    COALESCE(C.NOME, 'Sem Categoria') AS CATEGORIA  
FROM PRODUTO AS P  
LEFT JOIN CATEGORIA AS C ON P.ID_CATEGORIA = C.ID
```

SQL

- DQL: SELECT (MIN)

O MIN é uma **função de agregação** do SQL, que vai receber um único **parâmetro**.

Esse **parâmetro** deverá ser uma coluna da consulta, e a **função** vai comparar todos os valores daquela coluna a apresentar somente **o menor deles**.

SQL

- DQL: SELECT (MIN)

```
SELECT  
    MIN(P.VALOR)  
FROM PRODUTO AS P
```

SQL

- DQL: SELECT (MAX)

O MAX é uma **função de agregação** do SQL, que vai receber um único **parâmetro**.

Esse **parâmetro** deverá ser uma coluna da consulta, e a **função** vai comparar todos os valores daquela coluna a apresentar somente **o maior deles**.

SQL

- DQL: SELECT (MAX)

```
SELECT  
    MAX(P.VALOR)  
FROM PRODUTO AS P
```


SQL

- DQL: SELECT (COUNT)

O COUNT é uma função de agregação do SQL, que vai receber um único parâmetro.

Esse parâmetro deverá ser uma coluna da consulta, e a função vai contar quantos registros aquela coluna possui.

SQL

- DQL: SELECT (COUNT)

```
SELECT  
    COUNT(P.ID)  
FROM PRODUTO AS P
```

SQL

- DQL: **SELECT (SUM)**

O **SUM** é uma **função de agregação** do **SQL**, que vai receber um único **parâmetro**.

Esse **parâmetro** deverá ser uma coluna da consulta, e a **função** vai **somar todos os valores** daquela coluna.

SQL

- DQL: SELECT (SUM)

```
SELECT  
    SUM(P.VALOR)  
FROM PRODUTO AS P
```

SQL

- DQL: **SELECT**

Quando trabalhamos com **funções de agregação**, e além delas, queremos outras colunas sendo apresentadas na consulta, nós precisamos de mais alguns comandos.

Como essas **funções agregam os valores**, ou seja, elas **agrupam os registros**, as outras colunas que não estão usando essas **funções** também **precisam ser agrupadas**.

SQL

- DQL: **SELECT (GROUP BY)**

E para esses agrupamentos, nós temos o comando de **GROUP BY**.

Esse comando vai depois do **WHERE**, e caso não exista nenhum filtro na consulta, vai depois do último **JOIN**.

E da mesma maneira que o **ORDER BY**, é informado as colunas que serão agrupadas, todas separadas por **vírgula**.

SQL

- DQL: **SELECT (GROUP BY)**

As únicas colunas que não precisam ser informadas, são as colunas que fazem parte de alguma **função de agregação**.

SQL

- DQL: SELECT (GROUP BY)

```
SELECT  
  C.ID,  
  C.NOME,  
  COUNT(P.ID) AS QUANT_PRODUTOS  
FROM CATEGORIA AS C  
LEFT JOIN PRODUTO AS P ON P.ID_CATEGORIA = C.ID  
GROUP BY C.ID, C.NOME
```


Exercício

Exercício

gg.gg/SenacBD20

github.com/hmaltaurodev/slides