

AULA 11 – SELECT (PARTE 1)

Registros da tabela “cursos”

```
SELECT * FROM cursos;
```

Leia-se: “**Selecione** todas as **colunas** de **cursos**”

→ O símbolo de * significa “todas as colunas”

Ordenar colunas

```
ORDER BY nome;
```

Leia-se: “**Ordene** pelo **nome**”

→ Geralmente a ordem está sempre pela **chave primária**!

idcurso	nome	descricao	carga	totalaulas	ano
1	HTML5	Curso de HTML5	40	37	2014
2	Algoritmos	Lógica de Programação	20	15	2014
3	Photoshop5	Dicas de Photoshop CC	10	8	2014
4	PHP	Curso de PHP para iniciantes	40	20	2015
5	Java	Introdução à Linguagem Java	40	29	2015
6	MySQL	Bancos de Dados MySQL	30	15	2016
7	Word	Curso completo de Word	40	30	2016
8	Python	Curso de Python	40	18	2017
9	POO	Curso de Programação Orientada a Objetos	60	35	2016
10	Excel	Curso completo de Excel	40	30	2017

idcurso	nome	descricao	carga	totalaulas	ano
23	After Effects	Curso de Efeitos em Vídeos com After Effects	20	10	2018
2	Algoritmos	Lógica de Programação	20	15	2014
14	Android	Curso de Desenvolvimento de Aplicativos para ...	60	30	2018
13	C#	Curso de C#	30	12	2017
12	C++	Curso de C++ com Orientação a Objetos	40	25	2017
10	Excel	Curso completo de Excel	40	30	2017
18	Hardware	Curso de Montagem e Manutenção de PCs	30	12	2017
28	HTML4	Curso Básico de HTML, versão 4.0	20	9	2010
1	HTML5	Curso de HTML5	40	37	2014
5	Java	Introdução à Linguagem Java	40	29	2015

→ Eu ainda posso configurar pelo sentido!

```
ORDER BY nome DESC;
```

idcurso	nome	descricao	carga	totalaulas	ano
24	WordPress	Curso de Criação de Sites com WordPress	60	30	2019
7	Word	Curso completo de Word	40	30	2016
17	Swift	Curso de Desenvolvimento de Aplicativos para iOS	60	30	2019
21	SEO	Curso de Otimização de Sites	30	12	2017
20	Segurança	Curso de Segurança	15	8	2018
11	Responsividade	Curso de Responsividade	30	15	2018
19	Redes	Curso de Redes para Iniciantes	40	15	2016
8	Python	Curso de Python	40	18	2017
22	Premiere	Curso de Edição de Vídeos com Premiere	20	10	2017
16	PowerPoint	Curso completo de PowerPoint	30	12	2018

Ordem descendente

OBS: desc como **comando** significa **descreve (descreva)**, agora desc como parâmetro do **select** significa **descended (descendente)**

OBS: se eu quiser colocar na ordem ascendente, não preciso colocar nada, ou se quiser, usar o **asc**

Selecionando colunas

```
SELECT nome, carga, ano FROM cursos  
ORDER BY nome;
```

→ Neste comando, eu não estou selecionando todas as colunas

nome	carga	ano
After Effects	20	2018
Algoritmos	20	2014
Android	60	2018
C#	30	2017
C++	40	2017
Excel	40	2017
Hardware	30	2017
HTML4	20	2010
HTML5	40	2014
Java	40	2015

OBS: após o select, os nomes das colunas estarão em ordem! Eu posso escolher a ordem que quiser

ano	nome	carga
2010	HTML4	20
2010	PHP4	30
2014	HTML5	40
2014	Algoritmos	20
2014	Photoshop5	10
2015	PHP	40
2015	Java	40
2016	MySQL	30
2016	Word	40
2016	POO	60

Nota-se que no ano de 2014 a coluna nome não está ordenada!

```
select ano, nome, carga from cursos
order by ano;
```

ano	nome	carga
2010	HTML4	20
2010	PHP4	30
2014	Algoritmos	20
2014	HTML5	40
2014	Photoshop5	10
2015	Java	40
2015	PHP	40
2016	MySQL	30
2016	POO	60
2016	Redes	40

Agora está ordenado. Quando eu uso o comando **order by**, primeiro eu ordenei pela coluna ano e depois a coluna nome, nesta ordem, separados por vírgula

```
select ano, nome, carga from cursos
order by ano, nome;
```

Selecionando linhas

```
SELECT * FROM cursos
WHERE ano = '2016'
ORDER BY nome;
```

Leia-se: “Selecione todos os campos da tabela cursos onde o ano seja igual a 2016 ordenado pelo nome”

idcurso	nome	descricao	carga	totalaulas	ano
6	MySQL	Bancos de Dados MySQL	30	15	2016
9	POO	Curso de Programação Orientada a Objetos	60	35	2016
19	Redes	Curso de Redes para Iniciantes	40	15	2016
7	Word	Curso completo de Word	40	30	2016
NULL	NULL	NULL	NULL	NULL	NULL

→ Eu posso selecionar linhas e colunas específicas no mesmo comando!

nome	carga
MySQL	30
POO	60
Redes	40
Word	40

```
select nome, carga from cursos
where ano = '2016'
order by nome;
```

Aqui eu não estou mostrando a coluna nome, mas esta filtrada com os dados de

OBS: eu posso usar vários operadores relacionais (=, !=, <, >, <=, >=)

OBS: o símbolo de diferente (≠) representamos como != (não igual) ou então <> (maior menor)

Selecionando intervalos

```
SELECT * FROM cursos
WHERE totalaulas BETWEEN '20' AND '30'
ORDER BY nome;
```

Leia-se: “Selecione todos os campos da tabela cursos onde o total de aulas esteja entre 20 e 30 ordenada pelo nome”

nome	ano
MySQL	2016
POO	2016
Redes	2016
Word	2016
Java	2015
PHP	2015
Algoritmos	2014
HTML5	2014
Photoshop5	2014

```
select nome, ano from cursos
where ano between 2014 and 2016
order by ano desc, nome;
```

Selecionando valores

```
SELECT idcurso, nome FROM cursos
WHERE ano IN ('2014', '2016', '2018')
ORDER BY nome;
```

→ Qual a diferença entre o **IN** e o **BETWEEN**?

- ✓ No **IN** eu consigo colocar valores específicos
- ✓ No **BETWEEN** eu consigo especificar faixas de valores

nome	descricao	ano
HTML5	Curso de HTML5	2014
Algoritmos	Lógica de Programação	2014
Photoshop5	Dicas de Photoshop CC	2014
MySQL	Bancos de Dados MySQL	2016
Word	Curso completo de Word	2016
POO	Curso de Programação Orientada a Objetos	2016
Redes	Curso de Redes para Iniciantes	2016

```
select nome, descricao, ano from cursos
where ano in (2014, 2016)
order by ano;
```

Combinando testes

```
SELECT * FROM cursos
WHERE carga > 35 AND totaulas < 30
ORDER BY nome;
```

nome	carga	totaulas
PHP	40	20
Java	40	29
Python	40	18
C++	40	25
Redes	40	15
Magento	50	25
PHP7	40	20

```
select nome, carga, totaulas from cursos
where carga > 35 and totaulas < 30;
```

→ Devo lembrar sempre dos **OPERADORES LÓGICOS**!

p	q	p AND q
V	V	V
V	F	F
F	V	F
F	F	F

p	q	p OR q
V	V	V
V	F	V
F	V	V
F	F	F

Classificação do SELECT: DML

Para outros autores, acreditam que o select seja classificado como **Data Query Language (DQL)**

Ou seja, uma linguagem para perguntas, questionamentos