

Banco de Dados I- CSB30 - Grupo de Estudos

[Página inicial](#) / [Meus cursos](#) / [BD - csb - 2020](#) / Linguagens de Definição e Manipulação de Dados.

/ [Aula 06 - 15/09/2020 - Exercícios - SQL II](#)

Aula 06 - 15/09/2020 - Exercícios - SQL II

Considere as seguintes relações e as consultas abaixo.

Escolha, nos exercícios listados por último, 5 itens, e submeta o sql para eles em um arquivo pdf.

JOGADORES		
numj	integer	Numero do jogador (chave primária)
nome	char	Sobrenome
pnome	char	Primeiro nome
pais	char	Pais de origem
ano_n	integer	Ano de nascimento
ano_p	integer	Ano de profissionalizacao
cid_n	char	Cidade natal
cid_res	char	Cidade onde reside
tit_s	integer	Numero de titulos em simples
tit_d	integer	Numero de titulos em duplas
vits	integer	Numero de vitorias
derrs	integer	Numero de derrotas
natp	integer	Numero do jogador na ATP. Se nulo, o jogador nao esta ativo no momento.
LISTA_TORNEIOS		
numt	integer	Numero do torneio (chave primaria)
nomet	char	Nome do torneio
pais	char	Pais
cat	char	Categoria ('GS' Grand Slam e 'MS' Master Series)
quadra	char	Quadra ('D' dura, 'S' saibro, 'G' grama e 'T')
nparts	integer	Numero de participantes
TORNEIOS		
numt	number	Numero do torneio (chave estrangeira p/ torneios)



ano	number	Ano do torneio
numj	number	Numero do jogador que venceu o torneio (chave estrangeira para jogadores)
premio	number	Prêmio distribuído no torneio

Consultas de exemplo

0) Quais são as relações do banco de dados ---- SHOW TABLES

1) Obter todos os atributos de todas as tuplas das 5 relações.

SELECT *

FROM nome_da_relacao

2) Obter o primeiro nome, sobrenome, ano de nascimento e cidade residencial de todos os jogadores que nasceram no Brasil.

SELECT pnome, nome, ano_n, cid_res

FROM jogadores

WHERE pais = 'BRA'

3) Mesma consulta anterior, agora ordenando o resultado pelo primeiro nome dos jogadores.

SELECT pnome, nome, ano_n, cid_res

FROM jogadores

WHERE pais = 'BRA'

ORDER BY pnome

4) Obter, o ano e os vencedores dos torneios que aconteceram a partir do ano 1993 e que estejam ordenados por ano.

Note que a cláusula **"as"** permite renomear o nome de uma coluna

SELECT year **as** ano, winner **as** nome

FROM finals

WHERE year >= 1993

ORDER BY year

5) Obter, sem repetição e ordenados, os nomes de todas as cidades residenciais.

SELECT **DISTINCT** cid_res

FROM jogadores

ORDER BY cid_res

6) Nome dos jogadores que não moram **nem** em Las Vegas **nem** em Los Angeles.

SELECT nome

FROM jogadores

WHERE cid_res <> 'Las Vegas'

AND cid_res <> 'Los Angeles'

Note que nesta consulta um mesmo atributo foi comparado com mais de um valor constante. Neste caso pode-se utilizar o operador **IN**:

SELECT nome

FROM jogadores

WHERE cid_res **NOT IN** ('Las Vegas', 'Los Angeles')

7) Obter o nome dos jogadores que se tornaram profissionais com menos de 22 anos.

SELECT nome

FROM jogadores

WHERE **ano_p - ano_n < 22**

8) Obter o ano de nascimento do jogador mais velho

SELECT **MIN(ano_n)**

FROM jogadores

9) Obter o nome e ano de nascimento dos jogadores nascidos **entre** 1975 e 1981.

SELECT nome, ano_n

FROM jogadores

WHERE ano_n **BETWEEN** 75 and 81

10) Obter todos os dados dos jogadores que não estão ativos. Nesta consulta, precisamos das tuplas cujo atributo natp tem valor nulo (NULL). Quando um valor nulo é comparado através dos operadores "=" ou "<>", a comparação sempre falha. Verifique o resultado da consulta:

SELECT *

FROM jogadores

WHERE natp = NULL

Neste caso, precisamos utilizar o operador **"IS"**:

SELECT *

FROM jogadores

WHERE natp **IS** NULL

Para obter os dados dos jogadores que estão ativos fariamos:

```
SELECT *
FROM jogadores
WHERE natp IS NOT NULL
```

- 11) Obter o nome dos jogadores mais velhos que Kuerten. Devemos comparar cada tupla da relação jogadores com a tupla da relação jogadores correspondente ao jogador Kuerten. Neste caso, é necessário **criar "alias"** para a relação jogadores:

```
SELECT j1.nome
FROM jogadores j1, jogadores j2
WHERE j2.nome = 'Kuerten'
AND j1.ano_n < j2.ano_n
```

Outra maneira de fazer a consulta acima é utilizando uma subconsulta:

```
SELECT nome
FROM jogadores
WHERE ano_n < (SELECT ano_n
FROM jogadores
WHERE nome = 'Kuerten')
```

- 12) Para cada torneio ocorrido, obter seu nome, ano e numero do seu vencedor.

```
SELECT nome_t, ano, numj
FROM torneios, lista_torneios
WHERE torneios.numt = lista_torneios.numt
```

- 13) Obter o nome de cada jogador que venceu pelo menos um torneio.

```
SELECT nome
FROM jogadores
WHERE numj IN (SELECT numj
FROM torneios)
```

- 14) Obter o numero de torneios cadastrados.

```
SELECT COUNT(*)
FROM lista_torneios
```

- 15) Obter o numero de vezes em que realizaram-se torneios em um determinado pais.

```
SELECT pais, COUNT(*) as SEDES
FROM lista_torneios
GROUP BY pais
```

- 16) A consulta 10, em POSTGRES, pode ser escrita como abaixo. Experimente a sintaxe em Mysql

```
SELECT j1.nome
FROM jogadores as j1, jogadores as j2
WHERE j2.nome = 'Kuerten'
AND j1.ano_n < j2.ano_n
```

- 17) *para casa* Escreva as mesmas consultas 1 a 15 em álgebra e cálculo relacional

Exercícios

- Obter o nome de todos os jogadores exceto daquele cujo natp é 1271.
- Para cada jogador que venceu algum torneio obtenha o seu nome, o nome do torneio e o ano da vitória no torneio.
- Obter, sem repetições, o nome dos jogadores que já venceram algum torneio. Fazer esta consulta com uma junção.
- Obter os nomes dos jogadores que venceram uma ou mais vezes o torneio 1.
- Obter os nomes dos jogadores que não venceram torneios.
- Obter o(s) nome(s) do(s) jogador(es) mais velho(s) que todos os outros jogadores.
- Obter o nome de cada jogador que venceu o torneio de nome 'Wimbledon'.
- Obter o nome de cada jogador que não foi vencedor de torneios vencidos pelo jogador cujo nome é 'Sampras'.
- Fazer a consulta de exemplo 12 com uma junção.
- Obter a uma estatística de desempenho dos jogadores: porcentagem de vitórias de cada jogador em relação ao número total de jogos disputados. Apresentar os jogadores em ordem crescente dessa métrica.
- Obter o nome dos jogadores cujo número de vezes em que venceu um torneio seja maior ou igual a 2.

Status de envio

Status de envio	Nenhuma tentativa
Status da avaliação	Não há notas
Data de entrega	Monday, 23 Nov 2020, 05:08



Tempo restante A tarefa está atrasada há: 22 dias 6 horas

**Última
modificação** -

**Comentários
sobre o envio** ▶ [Comentários \(0\)](#)

◀ [Aula 06 - 15/09/2020 - SQL Inicial - Exercícios](#)

Seguir para...

[Exercício de SQL aninhado ▶](#)

Você acessou como Eduardo Vanderlei dos Santos Junior (Sair)

BD - csb - 2020

Português - Brasil (pt_br)

English (en)

Português - Brasil (pt_br)

Obter o aplicativo para dispositivos móveis