

Banco de Dados : **Junções**

Prof. Márcio Funes



Plano de aula

Introdução
Cláusula Join
Exercícios

Introdução

O contexto

No modelo relacional, as entidades tornar-se-ão tabelas. Onde cada tabela armazena dados distintintos dos demais.

Assim, um banco de dados relacional pode ter várias tabelas, de modo que os dados de um mesmo contexto ficam esparsos

O contexto

```
create table Cidades (  
    id            int            primary key,  
    nome          varchar(60)    not null,  
    populacao     int  
);  
  
create table Alunos (  
    id            int            primary key,  
    nome          varchar(60)    not null,  
    data_nasc     date,  
    cidade_id     int,  
  
    foreign key(cidade_id) references Cidades(id)  
);
```

```
insert into Cidades values (1, 'Arraial dos Tucanos' , 42632);  
insert into Cidades values (2, 'Springfield'      , 13820);  
insert into Cidades values (3, 'Hill Valley'       , 27383);  
insert into Cidades values (4, 'Coruscant'         , 19138);  
insert into Cidades values (5, 'Minas Tirith'      , 31394);  
  
insert into Alunos values ( 1, 'Immanuel Kant'    , '1724-04-22', 4);  
insert into Alunos values ( 2, 'Alan Turing'      , '1912-06-23', 3);  
insert into Alunos values ( 3, 'George Boole'     , '2002-01-01', 1);  
insert into Alunos values ( 4, 'Lynn Margulis'    , '1991-08-12', 3);  
insert into Alunos values ( 5, 'Nicola Tesla'     , '2090-01-08', null);  
insert into Alunos values ( 6, 'Ada Lovelace'     , '1978-05-28', 4);  
insert into Alunos values ( 7, 'Claude Shannon'   , '1982-10-15', 3);  
insert into Alunos values ( 8, 'Charles Darwin'   , '2010-02-06', null);  
insert into Alunos values ( 9, 'Marie Curie'      , '2007-07-12', 3);  
insert into Alunos values (10, 'Carl Sagan'       , '2000-11-20', 1);  
insert into Alunos values (11, 'Tim Berners-Lee'   , '1973-12-05', 4);  
insert into Alunos values (12, 'Richard Feynman'  , '1982-09-12', 1);
```

O contexto

Por outro lado, o usuário final precisa visualizar dados consolidados.

Contudo, consultas ao banco precisam visualizar os dados e suas relações.

Aluno	Data de Nascimento	Cidade
Immanuel Kant	1724-04-22	Coruscant
Alan Turing	1912-06-23	Hill Valley
George Boole	2002-01-01	Arraial dos Tucanos
Lynn Margulis	1991-08-12	Hill Valley
Nicola Tesla	2090-01-08	NULL
Ada Lovelace	1978-05-28	Coruscant
Claude Shannon	1982-10-15	Hill Valley
Charles Darwin	2010-02-06	NULL
Marie Curie	2007-07-12	Hill Valley
Carl Sagan	2000-11-20	Arraial dos Tucanos
Tim Berners-Lee	1973-12-05	Coruscant
Richard Feynman	1982-09-12	Arraial dos Tucanos

Cláusula Join

A cláusula JOIN

É possível usar o comando **SELECT** com mais de uma tabela na cláusula **FROM**.

Usa-se então o comando **JOIN** para fazer a **junção** de colunas de duas tabelas em uma seleção.

Combina dados de duas tabelas, normalmente a usa-se a **igualdade entre chaves** como critério.

O resultado da seleção a **intersecção** entre os registros conforme o critério estabelecido.

A cláusula JOIN

É possível usar o comando **SELECT** com mais de uma tabela na cláusula **FROM**.

Usa-se então o comando **JOIN** para fazer a **junção** de colunas de duas tabelas em uma seleção.

Combina dados de duas tabelas, normalmente a usa-se a **igualdade entre chaves** como critério.

Sintaxe típica:

```
select campos  
  from tabela1 join tabela2  
  on tabela1.pk = tabela2.fk;
```

Exemplo:

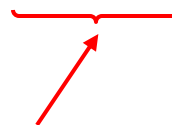
```
select *  
  from Alunos join Cidades  
  on Cidades.id = Alunos.cidade_id;
```


A cláusula JOIN

Exemplo:

```
1 • select *
2     from Alunos join Cidades
3     on Cidades.id = Alunos.cidade_id;
4
5
```

id	nome	data_nasc	cidade_id	id	nome	populacao
3	George Boole	2002-01-01	1	1	Arraial dos Tucanos	42632
10	Carl Sagan	2000-11-20	1	1	Arraial dos Tucanos	42632
12	Richard Feynman	1982-09-12	1	1	Arraial dos Tucanos	42632
2	Alan Turing	1912-06-23	3	3	Hill Valley	27383
4	Lynn Margulis	1991-08-12	3	3	Hill Valley	27383
7	Claude Shannon	1982-10-15	3	3	Hill Valley	27383
9	Marie Curie	2007-07-12	3	3	Hill Valley	27383
1	Immanuel Kant	1724-04-22	4	4	Coruscant	19138
6	Ada Lovelace	1978-05-28	4	4	Coruscant	19138
11	Tim Berners-Lee	1973-12-05	4	4	Coruscant	19138



Igualdade entre chaves estrangeira e primária

A cláusula JOIN

Há variações para a cláusula JOIN:

INNER JOIN - Retorna somente as linhas que casam em ambas as tabelas

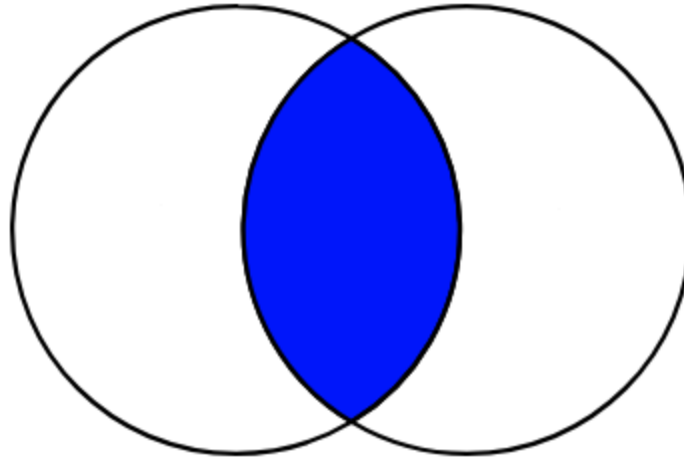
LEFT JOIN - Retorna todas as linhas da tabela da esquerda, e as linhas relacionadas da tabela da direita.

RIGHT JOIN - Retorna todas as linhas da tabela da direita, e as linhas relacionadas da tabela da esquerda.

A cláusula JOIN ou INNER JOIN

Por padrão, o resultado da seleção a intersecção entre os registros conforme o critério estabelecido.




Retorna somente as linhas que casam em ambas as tabelas



INNER JOIN

Retorna somente as linhas que casam em ambas as tabelas

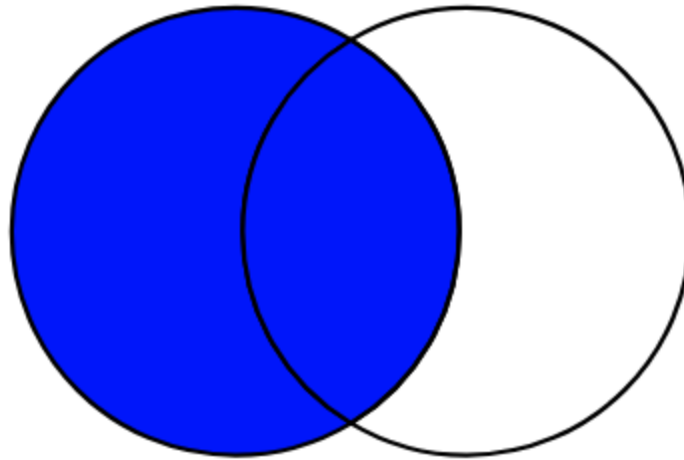
```
select *  
  from Alunos inner  
  join Cidades  
  on Cidades.id = Alunos.cidade_id;
```

<div><div> Filter Rows: <input type="text"/></div><div>Export:  Wrap Cell Content: </div></div>						
id	nome	data_nasc	cidade_id	id	nome	populacao
3	George Boole	2002-01-01	1	1	Arraial dos Tucanos	42632
10	Carl Sagan	2000-11-20	1	1	Arraial dos Tucanos	42632
12	Richard Feynman	1982-09-12	1	1	Arraial dos Tucanos	42632
2	Alan Turing	1912-06-23	3	3	Hill Valley	27383
4	Lynn Margulis	1991-08-12	3	3	Hill Valley	27383
7	Claude Shannon	1982-10-15	3	3	Hill Valley	27383
9	Marie Curie	2007-07-12	3	3	Hill Valley	27383
1	Immanuel Kant	1724-04-22	4	4	Coruscant	19138
6	Ada Lovelace	1978-05-28	4	4	Coruscant	19138
11	Tim Berners-Lee	1973-12-05	4	4	Coruscant	19138

LEFT JOIN

Retorna todos as linhas da tabela da esquerda, e as linhas relacionadas da tabela da esquerda.

Quando não há relacionamento entre os campos, os campos da tabela da direita serão NULL.



LEFT JOIN

Retorna todos as linhas da tabela da esquerda, e as linhas relacionadas da tabela da esquerda.

Quando não há relacionamento entre as linhas, os campos da tabela da direita serão NULL.

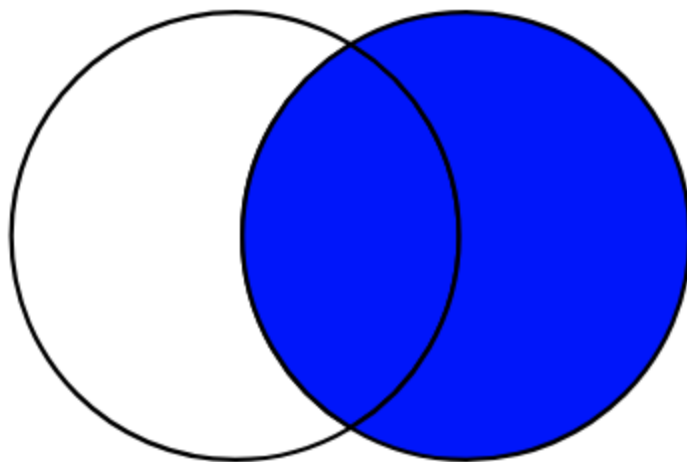
```
select *  
  from Alunos left join Cidades  
  on Cidades.id = Alunos.cidade_id;
```

Filter Rows: <input type="text"/> Export: <input type="button"/> Wrap Cell Content: <input type="button"/>						
id	nome	data_nasc	cidade_id	id	nome	populacao
1	Immanuel Kant	1724-04-22	4	4	Coruscant	19138
2	Alan Turing	1912-06-23	3	3	Hill Valley	27383
3	George Boole	2002-01-01	1	1	Arraial dos Tucanos	42632
4	Lynn Margulis	1991-08-12	3	3	Hill Valley	27383
5	Nicola Tesla	2090-01-08	NULL	NULL	NULL	NULL
6	Ada Lovelace	1978-05-28	4	4	Coruscant	19138
7	Claude Shannon	1982-10-15	3	3	Hill Valley	27383
8	Charles Darwin	2010-02-06	NULL	NULL	NULL	NULL
9	Marie Curie	2007-07-12	3	3	Hill Valley	27383
10	Carl Sagan	2000-11-20	1	1	Arraial dos Tucanos	42632
11	Tim Berners-Lee	1973-12-05	4	4	Coruscant	19138
12	Richard Feynman	1982-09-12	1	1	Arraial dos Tucanos	42632

RIGHT JOIN

Retorna todos as linhas da tabela da direita, e as linhas relacionadas da tabela da esquerda.

Quando não há relacionamento entre os campos, os campos da tabela da esquerda serão NULL.



RIGHT JOIN

Retorna todos as linhas da tabela da direita, e as linhas relacionadas da tabela da esquerda.

Quando não há relacionamento entre os campos, os campos da tabela da esquerda serão NULL.

```
select *
  from Alunos right join Cidades
 on Cidades.id = Alunos.cidade id;
```

Filter Rows:

Export:

Wrap Cell Content:

id	nome	data_nasc	cidade_id	id	nome	populacao
3	George Boole	2002-01-01	1	1	Arraial dos Tucanos	42632
10	Carl Sagan	2000-11-20	1	1	Arraial dos Tucanos	42632
12	Richard Feynman	1982-09-12	1	1	Arraial dos Tucanos	42632
NULL	NULL	NULL	NULL	2	Springfield	13820
2	Alan Turing	1912-06-23	3	3	Hill Valley	27383
4	Lynn Margulis	1991-08-12	3	3	Hill Valley	27383
7	Claude Shannon	1982-10-15	3	3	Hill Valley	27383
9	Marie Curie	2007-07-12	3	3	Hill Valley	27383
1	Immanuel Kant	1724-04-22	4	4	Coruscant	19138
6	Ada Lovelace	1978-05-28	4	4	Coruscant	19138
11	Tim Berners-Lee	1973-12-05	4	4	Coruscant	19138
NULL	NULL	NULL	NULL	5	Minas Tirith	31394

Para saber mais

- https://www.w3schools.com/sql/sql_join.asp
- <https://dev.mysql.com/doc/refman/5.7/en/join.html>
- <https://www.techonthenet.com/mysql/joins.php>
- <https://www.mysqltutorial.org/mysql-join/>
- <https://www.guru99.com/joins.html>
- <https://www.w3resource.com/mysql/advance-query-in-mysql/mysql-joins.php>
- <https://www.sqlfromhell.com/entendendo-join-sql/>

Busquem conhecimento



Exercícios



Exercícios

1. Selecione o nome e a espécie de todos os animais
2. Selecione todos os gatos laranja
3. Selecione todos os cachorros da cor amarelo
4. Selecione todos os animais aquáticos que pesam mais que 70 Kg
5. Selecione todos os herbívoro ordenados pelos mais pesados
6. Selecione todos os carnívoro que são pretos e brancos
7. Selecione todos os onívoros que nasceram antes de 2013
8. Selecione todos os mamíferos que pesam mais que 10 quilos e são marrons ou azul
9. (Desafio) Selecione a quantidade total de animais
10. (Desafio) Se somarmos os peso de todos os gatos, qual será o resultado?