Banco de Dados Relacionais

 Consultas SQL Avançadas

// Prática Individual

# Primera Parte

Responda às seguintes questões:

1. Como é chamado um **JOIN** em um banco de dados e para que é usado?

O JOIN é uma forma de combinar e consultas os dados quando eles estão em diferentes tabelas

1. Explique os tipos de **JOIN**.

Temos 5 tipos de JOIN: o INNER JOIN é o mais comum e mais usado, sendo caracterizado pela junção dos dados de 2 tabelas que possuem algo em comum, como uma chave estrangeira; O LEFT JOIN é usado para pegar todos os dados da tabela que está a esquerda, ou seja antes da instrução LEFT JOIN incluindo os registros correspondentes da outra tabela, pegando até mesmo os dados nulos; Já o RIGHT JOIN é o extremo oposto do LEFT JOIN, então ele pega os dados da tabela que está a direita da instrução; O FULL JOIN retorna todos os registros que tiverem alguma correspondência em uma ou na outra tabela; Por ultimo temos o CROSS JOIN que faz o produto cartesiano entre as tabelas, ou seja, trazendo todos os dados que tiverem correspondência nas duas tabelas.

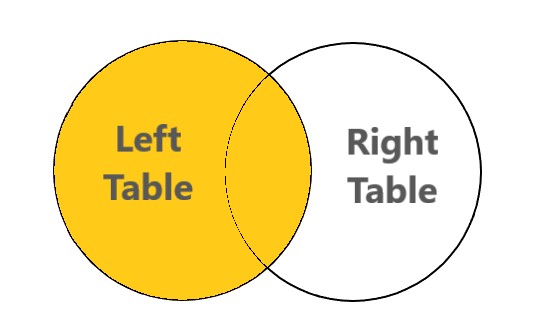
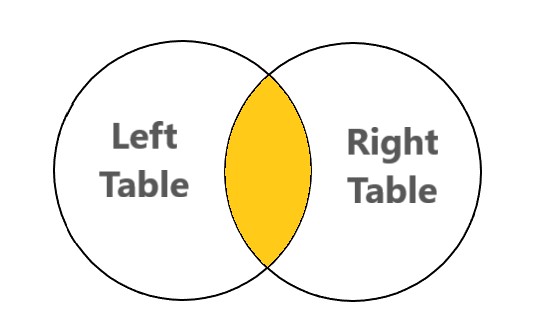
1. Para que serve o **GROUP BY**?

Serve para agrupar os registros por um atributo em comum.

1. Para que serve o **HAVING**?

Ele serve como um WHERE, só que deve ser usado quando queremos ter uma condição que utilize uma função de agregaçãol

1. Escreva uma consulta genérica para cada um dos seguintes diagramas:



Select tabela1.nome, tabela2.nome from tabela1 inner join tabela2;

Select tabela1.nome, tabela2.nome from tabela1 left join tabelo2;

# Segunda Parte

Faça as seguintes consultas ao banco de dados **movies\_db.sql** trabalhado na primeira aula.

1. Mostrar o título e o nome do gênero de todas as séries.

select title, name from series inner join genres where genre\_id = genres.id;

1. Mostrar o título dos episódios, o nome e sobrenome dos atores que trabalham em cada um deles.

select title, first\_name, last\_name from episodes inner join actors where actors.id in

(select actor\_id from actor\_episode where episode\_id = episodes.id);

1. Mostrar o título de todas as séries e o número total de temporadas que cada uma delas possui.

select series.title, count(\*) as temporadas from seasons inner join series on series.id = seasons.serie\_id group by series.id;

1. Mostrar o nome de todos os gêneros e o número total de filmes de cada um, desde que seja maior ou igual a 3.

select count(movies.title) as filme, genres.name as genero from genres inner join movies on movies.genre\_id = genres.id group by genres.name;

1. Mostrar apenas o nome e sobrenome dos atores que atuam em todos os filmes de

Star Wars e que estes não se repitam.

select first\_name, last\_name from actors where actors.id in (select actor\_id from actor\_movie where actors.id = actor\_id and movie\_id in

(select movies.id from movies where title like "%Star Wars%"));

Como guia para construir consultas, lembre-se do DER do cenário:

