

WebDev - Módulo 4

Roteiro de aula

Aula 8 - Hard: Mais informações



Tópicos da aula:

- Agregação + filtragem
- HAVING
- Paginação
- CASE/WHEN



Objetivos da aula:

1. Executar a criação de consultas;
2. Descrever o mecanismo de agregação e sua aplicação;
3. Expressar a ordem de execução das consultas;
4. Executar consultas a partir de diversas operações;
5. Descrever o conceito de paginação, definindo quando deve ser usado;
6. Descrever execução do CASE e aplicar em consultas;



Atividades da Aula

→ Atividade: Aprofundando

- ◆ Vocês vão ficar surpresos em saber que temos mais perguntas para responder. Agora, vocês deverão criar as consultas que respondem às seguintes perguntas:

- Qual o ranking dos DVDs que foram mais alugados?
- Quantas vezes o filme com id = 1 na loja 1 foi alugado?
- Qual funcionário alugou mais filmes?
- Qual o faturamento do dia 27 de maio de 2005?
- Quanto cada funcionário faturou no dia 27 de maio de 2005?"

→ Atividade: Classificados

- ◆ Chegou um ticket para classificarmos os nossos clientes e os atores dos nossos filmes: nossos clientes serão classificados pelos seus gastos e os atores pelo número de participações em filmes. Criem consultas

considerando as classificações abaixo:

- Classificar os clientes por faixa de gastos:
 - Acima de 100\$ - 'Clientão brabo'
 - Entre 70\$ e 100\$ - 'Cliente bom'
 - Abaixo de 70\$ - 'Cliente eventual'
- Classificar os atores por número de participações:
 - Acima de 40 participações - 'Topa tudo'
 - Acima de 30 participações - 'Altamente produtivo'
 - Acima de 20 participações - 'Fez alguns filmes'
 - Acima de 10 participações - 'Tem que fazer seu nome ainda'



Momento Aprendizagem em Pares

- Esse momento é dedicado para vocês desenvolverem suas demandas e entregas para o curso.
- Utilize esse tempo da maneira que preferir, mas atente-se às aulas que você deve realizar as entregas.

- ◆ **Dica:** Nas Propostas dos projetos, vocês encontram uma sugestão de organização para a realização das atividades.
- ◆ **Lembre-se:** O momento de Aprendizagem em Pares é justamente para fazer trocas e aprender em comunidade, aproveite seus colegas e se desenvolvam juntos!

→ Entregas:

- ◆ Projeto individual: aula 5 - HARD
- ◆ Projeto em grupo: aula 10 - HARD
- ◆ Apresentação do projeto: aula 10 - HARD



Revisão da aula

- Os **filtros** SQL são strings de texto que você usa para especificar um subconjunto dos itens de dados em um tipo de dados de banco de dados SQL ou interno. Para banco de dados SQL e em tipos de dados internos, o filtro é uma cláusula SQL **WHERE** que fornece um conjunto de comparações que devem ser verdadeiras para que um item de dados seja retornado.
- Uma função de **agregação** em SQL executa um cálculo em vários valores e retorna um único valor. O SQL fornece muitas funções de agregação que incluem avg, count, sum, min, max, etc. Uma função de agregação ignora valores NULL quando executa o cálculo, exceto para a função de contagem.

- A cláusula **HAVING** no MySQL também é usada para filtrar os dados, assim como a cláusula where. HAVING no MySQL **filtra as linhas do conjunto de resultados intermediário** que é construído usando as cláusulas FROM, WHERE ou GROUP BY em uma instrução SELECT.
- HAVING no MySQL é normalmente usada com uma cláusula **GROUP BY**. Isso significa que a cláusula **HAVING** é usada em combinação com uma cláusula GROUP BY para restringir o número de grupos a serem retornados satisfazendo a condição especificada usando a HAVING.
- A **paginação** é um conceito muito interessante no MySQL e pode ser feito facilmente pelas cláusulas **LIMIT** e **OFFSET**. É muito **útil quando você deseja exibir o grande conjunto de registros em várias páginas**.
- A expressão **MySQL CASE** é uma estrutura de fluxo de controle que **permite adicionar lógica if-else a uma consulta**. De um modo geral, você pode usar a expressão CASE em qualquer lugar que permita uma expressão válida, por exemplo, cláusulas SELECT, WHERE e ORDER BY.
- A expressão **CASE** tem duas formas: **CASE simples** e **CASE pesquisado**.
- Observe que o MySQL tem uma instrução **CASE** que você pode usar apenas em programas armazenados, como procedimentos armazenados, funções armazenadas, eventos e gatilhos.
- O seguinte exemplo ilustra a sintaxe de uma expressão CASE simples:
 - CASE value
 - WHEN valor1 THEN resultado1
 - WHEN valor2 THEN resultado2
 - ...
 - [ELSE else_resultado]
 - END
- Nesta sintaxe, **CASE** combina o valor com o valor1, valor2, etc., para igualdade e retorna o resultado1, resultado2,... Se o valor não for igual a nenhum valor1, valor2, ... **CASE** retorna o resultado na cláusula **ELSE** se a cláusula **ELSE** é especificada.

→ O **CASE** compara o valor com valores nas cláusulas **WHEN** para igualdade, você não pode usar com **NULL** porque NULL = NULL retorna false.



Para ajudar



Links interessantes:

- Para aprender mais sobre SQL HAVING:
https://www.w3schools.com/sql/sql_having.asp
- Para aprender mais sobre SQL CASE:
https://www.w3schools.com/sql/sql_case.asp
- Tutorial de group do SQL:
<https://www.freecodecamp.org/portuguese/news/tutorial-de-group-do-sql-count-sum-average-e-instrucao-having-explicadas/>