EXERCICIO FIXAÇÃO: UNIDADE 12

Crie um projeto ou uma pasta **BancoDeDados.**

1. Crie uma tabela Aluno com as colunas Nome (VARCHAR(255)), Email (VARCHAR(255)) Telefone (VARCHAR(10)), Altura (DECIMAL(3,2)) e Aprovado (TINYINT(1)).
2. Insira alguns registros na tabela evitando valores repetidos.
3. Utilizando a cláusula **WHERE** faça uns exemplos criando uma consulta para cada tipo de operador. EX:

SELECT \* FROM Aluno WHERE Altura = 1.8;

SELECT \* FROM Aluno WHERE Altura != 1.8;

SELECT \* FROM Aluno WHERE Altura <= 1.8;

SELECT \* FROM Aluno WHERE Altura < 1.8;

SELECT \* FROM Aluno WHERE Altura >= 1.8;

SELECT \* FROM Aluno WHERE Altura > 1.8;

SELECT \* FROM Aluno WHERE Aprovado IS TRUE ;

SELECT \* FROM Aluno WHERE Aprovado IS NOT TRUE ;

SELECT \* FROM Aluno WHERE Nome IS NULL ;

SELECT \* FROM Aluno WHERE Nome IS NOT NULL ;

SELECT \* FROM Aluno WHERE Altura BETWEEN 1.5 AND 1.8;

SELECT \* FROM Aluno WHERE Altura NOT BETWEEEN 1.5 AND 1.8;

SELECT \* FROM Aluno WHERE Nome LIKE ’Rafael %’;

SELECT \* FROM Aluno WHERE Nome NOT LIKE ’Rafael %’;

SELECT \* FROM Aluno WHERE Altura IN (1.5 , 1.6 , 1.7 , 1.8) ;

SELECT \* FROM Aluno WHERE Altura NOT IN (1.5 , 1.6 , 1.7 , 1.8) ;

SELECT \* FROM Aluno WHERE NOT Altura = 1.80;

SELECT \* FROM Aluno WHERE Altura < 1.8 AND Nome LIKE ’Nome ’;

SELECT \* FROM Aluno WHERE Altura < 1.8 OR Nome LIKE ‘Nome%’;

Comente o que cada linha faz.

Obs: nas consultas utilize valores que façam sentido de acordo com os valores que você inseriu na tabela.

1. Utilizando a tabela Aluno crie uma consulta que traga todos os alunos, sendo que os primeiros devem ser listados os alunos aprovados e em seguida os reprovados. EX:

SELECT \* FROM Aluno ORDER BY Aprovado DESC; (Ordena decrescente)

1. Utilizando a tabela Aluno crie uma consulta que traga todos os alunos aprovados (WHERE Aprovado = 1) ordenados pelo e-mail. Para desempate utilize as colunas altura (na ordem decrescente) e depois nome.
2. Utilizando a tabela Aluno crie uma consulta calcule a soma das alturas dos alunos reprovados. Além da soma, faça uma consulta que conte o total de registros, o maior e menor altura e quantos aprovados.

Lembrete: As tabelas e as colunas consultadas podem ser “apelidadas”. Esse recurso pode facilitar a criação das consultas e a análise dos resultados. Para isso, devemos utilizar o comando AS.

Ex: SELECT a. nome AS aluno\_nome FROM Aluno AS a

Obs: Veja que podemos adicionar dados repetidos, tirando o Id, no banco de dados sem problema algum. E no caso de um campo CPF podemos ter o mesmo número? Por isso existe uma função de índices exclusivos: EX: UNIQUE NONCLUSTERED ([Campo] ASC)

1. Crie uma tabela Livro para que ela contenha, pelo menos, as colunas Titulo e Autor, ambas VARCHAR(255).
2. Insira alguns valores na tabela Livro. Tente inserir valores com títulos e autores repetidos. Tente inserir valores com títulos e autores repetidos e observe os resultados.
3. Remova todos os registros da tabela Livro e faça uma alteração na mesma para que não seja permitida a inserção de valores repetidos nas colunas Titulo e Autor. Tente inserir valores com Títulos e Autores repetidos e observe os resultados.
4. Façam um CRUD da tabela livro, caso queiram faça uma junção com outra tabela.

Bons Estudos!