

BANCO DE DADOS

EXERCÍCIOS – SPRINT I

1 – Loja de Jogos

No MySQL Workbench, crie o banco de dados "game_store".

A loja de jogos "Game Store" estava passando dificuldades de organização pois o dono Mário, um simpático senhor italiano aposentado que trabalhou com encanamento durante sua juventude, não tinha controle do seu estoque de jogos.

Para ajudar Mário a organizar sua loja de jogos, um estudante da SPTech propôs criar uma tabela chamada "jogo" que continha o **id** como número e chave primária que se autoincrementa, o **nome** do jogo com tamanho máximo de 30 caracteres, o nome do **diretor** do jogo com tamanho máximo de 30 caracteres, o **gênero** com tamanho de 30 caracteres, a data de **lançamento**, a **nota** do jogo (entre 0-10) e um campo para guardar a **quantidade** de unidades disponíveis.

1. Insira pelo menos 5 registros para a tabela.
2. Altere a tabela para inserir uma coluna que represente o tipo de mídia que deve armazenar o tipo de jogo apenas com os valores "física" ou "digital".
3. Atualize os registros dos jogos inseridos anteriormente.
4. Exiba apenas os jogos com data de lançamento a partir de 2015.
5. Exiba os jogos que tenham a letra "a" em seu nome e são de mídia física.
6. Exiba os jogos onde o nome do diretor não contenha a letra "e".
7. Altere a tabela para criar uma checagem na inserção da nota, onde o valor não pode ser menor que 0 e nem maior que 10.
8. Exiba os jogos de um determinado gênero e que ainda esteja em estoque.
9. Exclua os jogos que não têm mais unidades disponíveis em estoque.
10. Renomeie a coluna "diretor" para "criador" e exiba como ficou a estrutura da tabela.

2 – Esportes Olímpicos

No MySQL Workbench, crie o banco de dados "olimpiadas".

Casimiro, um fã apaixonado por esportes, decidiu que queria acompanhar de perto as Olimpíadas. Ele pediu ajuda a um amigo para criar uma tabela que armazenasse informações sobre os esportes praticados nas Olimpíadas.

Eles criaram uma tabela chamada "esporte" que continha **id** como número e chave primária que se autoincrementa, **nome** que armazena o nome do esporte com limite de 40 caracteres, a **categoria** do esporte com limite de 20 caracteres que aceita apenas ("Individual" ou "Coletivo"), o **número de jogadores** que podem participar por equipe, **estreia** que armazena a data de quando o esporte estreou nas Olimpíadas e o nome do **país de origem** do esporte com até 30 caracteres.

1. Insira pelo menos 5 registros na tabela.
2. Altere a tabela para adicionar uma coluna popularidade que armazene a popularidade do esporte como um valor decimal entre 0 e 10 e exiba como ficou a estrutura da tabela.
3. Atualize os registros para definir a popularidade dos esportes inseridos anteriormente.
4. Exiba os esportes ordenados por popularidade em ordem crescente.
5. Exiba apenas os esportes que estrearam nas Olimpíadas a partir do ano 2000.
6. Crie uma checagem para que não possa ser inserido valores dentro de estreia que seja menor que 06 de abril de 1896 e depois da data atual.
7. Altere a tabela para excluir a regra de inserção de categoria, assim podendo colocar valores além de "Individual" ou "Coletivo".
8. Exiba apenas os esportes cujo nome do país de origem tenha "a" na segunda letra.
9. Exiba os dados onde o número de jogadores por equipe esteja entre 4 e 11.
10. Remova os registros onde id seja 1, 3 e 5.

3 – Desenhos Animados

No MySQL Workbench, crie o banco de dados "desenho".

Você foi contratado para criar um banco de dados que armazene informações detalhadas sobre desenhos animados exibidos na emissora. O objetivo é manter um catálogo com as seguintes informações.

Para atender a essa necessidade, você propôs a criação de uma tabela com os seguintes campos: **id**, um número inteiro chave primária da tabela, este valor deverá ser autoincrementado a partir do 10. **Título**, um campo de texto com tamanho de 50 caracteres que armazenará o título do desenho animado. A **data de lançamento** um campo que armazenará a data de lançamento do desenho. O nome da **emissora original**, um campo de texto com tamanho de 50 caracteres. **Classificação**, um campo que armazenará a classificação indicativa do desenho. **Status**, um campo de texto com tamanho de 15 caracteres, que armazenará o status do desenho. **Nota**, um campo que armazenará a avaliação do desenho, permitindo valores de 1 a 5.

1. Exibir todos os dados da tabela
2. Exibir todos os desenhos com a classificação menor ou igual a 14 anos
3. Exibir todos os desenhos de uma mesma emissora original
4. Modificar o campo status, para que aceite apenas o status 'exibindo', 'finalizado', 'cancelado', tente inserir algum outro valor para ver se a regra foi aplicada
5. Modificar o status 'exibindo' para 'finalizado' de 2 desenhos pelo ID
6. Deletar a linha do desenho de ID 12
7. Exibir apenas os desenhos que comecem com uma determinada letra
8. Renomear a coluna classificacao para classificacaoIndicativa
9. Atualizar a nota e data de lançamento do desenho de ID 11
10. Limpe todos os dados da tabela
11. Remover a regra do status do desenho

4 – Despensa do Scooby-Doo

No MySQL Workbench, crie o banco de dados "estoque".

A turma do Scooby Doo percebeu que os alimentos guardados no armário estavam sumindo de forma muito rápida. Para ter um melhor controle, Fred propôs criar um banco de dados que registrasse os dados de entrada e saída dos alimentos do armário.

Ele criou uma tabela chamada "MisteriosSA", que possui os campos **id** um número inteiro chave primária, **nome** um campo de texto para armazenar o nome do produto, **data da compra** para armazenar a data na qual o produto foi comprado, **preço** valor numérico com casas decimais, **peso** um valor numérico que armazena o peso em gramas e **data de retirada** para armazenar a data em que o produto foi consumido.

1. Insira na tabela, no mínimo 5 compras de alimentos com datas diferentes. Por agora, não preencha a coluna referente a "data de retirada".
2. Verifique se os valores foram inseridos corretamente.
3. Exiba os nomes, as datas de compra e retirada e o id dos alimentos ordenados a partir da data de compra mais antiga.
4. Alguém comeu uma caixa de biscoitos, atualize a data de retirada da caixa de "Biscoitos Scooby" que foi comprada a mais tempo.
5. Altere o nome da coluna id para idComida.
6. Altere o tipo do check para que os alimentos só possam ser "Biscoitos Scooby" ou "Cachorro-quente".
7. Exiba os produtos onde o nome seja "Biscoitos Scooby" de forma que o nome das colunas dataCompra apareça como "data da compra" e dataRetirada apareça como "data da retirada".
8. Exiba os alimentos que foram comprados antes do dia 25 de julho de 2024.
9. Exiba os alimentos que possuem um preço acima ou igual a 30.50.
10. Limpe a tabela.

5 – Heróis

No MySQL Workbench, crie o banco de dados “vingadores”.

Crie uma tabela chamada “herói”, onde terá um campo de **nome** contendo até 45 caracteres. O seu herói poderá ter uma **versao**, onde será armazenado o filme de sua escolha, esse campo aceitará até 45 caracteres. Todo herói, tem uma **habilidade** que se destaca entre os outros super-heróis. Todo herói tem uma **altura**, esse campo aceitará valores inteiros representando centímetros. O id deverá se autoincrementar.

Insira 5 heróis para derrotar o vilão **Doutor Destino**.

1. Exiba os dados inseridos na tabela
2. Adicione um campo de regeneração, onde ele aceitará apenas os valores booleanos de TRUE ou FALSE.
3. Modifique o campo versão para aceitar até 100 caracteres.
4. Remova o herói de id 3 pois ele se morreu em batalha.
5. Chegou reforços, insira um novo herói para a equipe.
6. Exiba todos os dados inseridos na tabela onde o nome do herói começa com “C” ou “H”.
7. Exiba todos os dados inseridos na tabela onde o nome do herói não contém a letra “A” no campo nome.
8. Exiba apenas o nome do herói onde a altura for maior que 190.
9. Exiba todos os dados da tabela de forma decrescente pelo nome onde a altura do herói for maior que 180.
10. Limpe os dados da tabela.