




Banco de dados I


Projeto

Instruções

 Envios até **30/09**, segunda-feira, às **23h59**.


 Submissão no LMS

Todos devem submeter o trabalho, independente do trabalho ter sido em grupo!

 Formato de envio: pasta zipada contendo 3 arquivos:

1. Diagrama com o modelo relacional (parte 1)
2. Arquivo .sql com todos os comandos DDL, DML e DQL com o modelo (parte 1)
3. Arquivo .sql com as consultas que respondem a parte 2

Importante : **identificar o zip com o número do grupo**, conforme definido aqui para que eu possa identificar os integrantes (ex. grupo1.zip)

 **Parte 1:** escolher um dos projetos

1. Criar o modelo e aplicar as formas normais
2. Criar o banco de dados (DDL)
3. Inserir ao menos 5 registros em cada tabela (DML)
4. Atualizar e deletar ao menos 2 registros (DML)
5. Realizar 4 consultas (1 com agrupamento, 3 com joins) (DQL)

Parte 2: usar o banco de dados já populado para responder as perguntas por meio de consultas (DQL)

Parte 1 – Projeto 1

Sistema de Gerenciamento de Restaurante

O restaurante ada.food deseja organizar informações para melhorar sua logística e funcionamento, incluindo dados sobre seus clientes, funcionários, pratos, pedidos, mesas e reservas. Seguem os requisitos:

- Cada cliente deve ter um registro contendo nome, cpf, telefone, email e data de nascimento.
- Um cliente pode fazer reservas e pedidos.
- Cada funcionário deve ter um registro contendo nome, cpf, cargo (ex.: Garçom, Cozinheiro), e telefone.
- Funcionários podem estar envolvidos em diferentes pedidos, e cada pedido pode ser atendido por vários funcionários (ex.: garçom que atende e cozinheiro que prepara o prato).
- O restaurante oferece vários pratos, cada um com um nome, descrição, e preço.
- O sistema deve permitir o registro de pedidos, incluindo a data, hora, e o cliente que fez o pedido.
- Um pedido pode incluir vários pratos, e cada prato pode estar em vários pedidos.
- O restaurante possui várias mesas, cada uma com um número identificador e capacidade de assentos.
- Clientes podem fazer reservas de mesas, incluindo a data, hora e o número da mesa reservada.

Parte 1 – Projeto 2

Sistema de Gerenciamento de Campeonato de Artes Marciais

A federação de artes marciais ada.fight deseja organizar os dados dos campeonatos que realiza. O banco de dados deve permitir o registro de atletas, equipes, categorias de peso, lutas, resultados, e a gestão de eventos de campeonatos.

- Cada atleta deve ter um registro contendo nome, data de nascimento, graduação (ex.: faixa-preta, faixa-marrom), e a equipe à qual pertence.
- Cada atleta compete em uma determinada categoria de peso.
- O sistema deve armazenar informações sobre as equipes de artes marciais, incluindo nome da equipe e cidade de origem.
- As categorias de peso devem ser registradas, incluindo o nome da categoria (ex.: Leve, Médio, Pesado) e o limite de peso (em kg).
- O sistema deve permitir o registro de campeonatos, incluindo nome, data de início, data de término e local.
- As lutas devem ser registradas, associando dois atletas, a categoria de peso, a data e o campeonato em que a luta ocorreu.
- O resultado da luta deve ser registrado (ex.: Vitória, Derrota, Empate, Finalização).
- O sistema deve registrar os resultados de cada luta, incluindo o vencedor, a técnica utilizada (ex.: finalização, pontos) e o tempo da luta.

Parte 2

Tendo como base o banco de dados dvd_rental disponível aqui, elabore as consultas SQL necessárias para responder às seguintes perguntas:

1. Quais são os filmes (film.title), suas respectivas categorias (category.name) e idioma (language.name)?
2. Quais foram os filmes alugados entre os dias 1 de fevereiro e 15 de março de 2006 na loja 1 (store.id = 1).
3. Qual a quantidade de vezes que cada filme foi alugado?
4. Para cada funcionário (staff.staff_id), qual é o valor total retornado em locações (payment.amount) e qual é a quantidade total de locações (rental.rental_id)?
5. Qual é o nome de categoria de filme (category.name) que nos retorna mais dinheiro no total (payment.total)?

Ordene sua consulta para mostrar o resultado em ordem de lucratividade, do nome de categoria mais lucrativo para o nome de categoria menos lucrativo e garanta que todos os nomes de categorias apareçam no resultado final, inclusive aquelas que não tiverem nenhuma locação associada a elas.

[Detalhes do banco de dados aqui.](#)