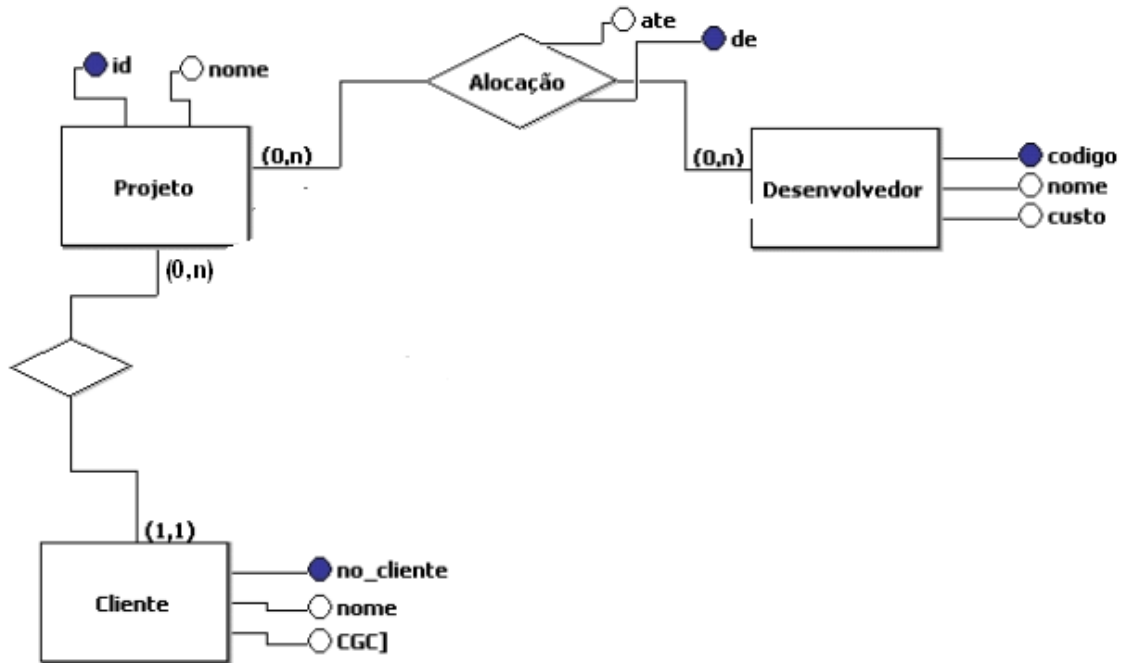


**Banco de Dados - Lista 5 – MER- MR - SQL**

1) A partir do modelo entidade-relacionamento abaixo:

- Elabore o mapeamento relacional.
- Crie as tabelas no banco de dados.
- Insira dois registros em cada tabela.



Considere as seguintes regras:

- A escolha do tipo e tamanhos dos campos fica a critério do(a) aluno(a).
- Todos os atributos são obrigatórios
- O nome do projeto também é único
- O custo do desenvolvedor deve ser maior que 0.
- O CGCJ também é único

2) Elabore o modelo entidade relacionamento considerando os requisitos descritos abaixo:

Contexto: a aplicação deve facilitar o gerenciamento das informações relacionadas a um jogo educativo. Vocês devem lembrar que algumas regras de negócio não são representadas no MER. Os requisitos listados representam uma simplificação do contexto real.

RF1: o s.d permitir manter informações sobre os jogadores contendo os seguintes dados: código(único), login (único), senha (única), nome e email (pode ser mais de um).

RF2: o s.d permitir manter informações sobre os jogos contendo os seguintes dados: código (único), descrição e duração máxima (maior que 0).

RF3: Um jogo pode conter várias etapas e uma etapa deve ser obrigatoriamente de um e somente um jogo. Cada etapa contém um número e a descrição. Este número pode repetir para jogos diferentes. Contudo, para um mesmo jogo, as etapas devem possuir números diferentes.

Um jogo tem, no mínimo, 1 ou várias etapas. Quando um jogo é excluído, todas as etapas também devem ser removidas.

RF4: um jogador pode jogar vários jogos e um jogo pode ser jogado por vários jogadores. Contudo, para cada jogador em um jogo, é necessário manter o resultado.

RF5: O s.d permitir manter informações sobre os prêmios concedidos contendo os seguintes dados: código (único) e gratificação. Um jogador pode ganhar um mesmo prêmio em vários jogos. Um prêmio em um jogo pode ser dado a somente um jogador. Um jogador em um jogo pode ganhar vários prêmios. Um jogador pode ganhar o mesmo prêmio em vários jogos. É necessário manter a data que o jogador ganhou o prêmio em um determinado jogo.

RF6: um jogador pode ser mestre de nenhum ou mais jogadores. Além disso, um jogador pode ter nenhum ou um mestre.

- 3) Elabore o mapeamento para o modelo relacional considerando o MER da questão 2.
- 4) Escreva o código necessário para implementação das tabelas definidas na questão anterior.
- 5) Escreva o código necessário para inserção de 2 registros em cada tabela criada na questão anterior.

6) Imagine que o RF4 seja alterado para a seguinte regra:

RF4: um jogador pode jogar vários jogos e um jogo pode ser jogado por vários jogadores ao mesmo tempo. Contudo, para cada jogador em um jogo, é necessário manter a data de início, data de fim e resultado. Além disso, um jogador não pode jogar o mesmo jogo mais de uma vez com a mesma data de início.

Qual impacto desta alteração no esquema conceitual, lógico e físico do banco de dados?

7) Elabore o modelo conceitual e o mapeamento para o modelo lógico considerando uma aplicação que deverá gerenciar informações sobre um site de treinamento em conteúdos diversos. Os treinamentos são realizados a partir dos cursos oferecidos. Depois de elaborar o mapeamento, implemente o banco de dados e insira, no mínimo, duas tuplas em cada tabela. É importante lembrar que as informações abaixo representam uma situação hipotética e uma simplificação de um contexto real.

- Todo usuário deve ter um login que o identifica, uma senha, o nome e o(s) endereço(s) eletrônico(s). Existem dois tipos de usuário: aprendiz e tutor. Para o tutor, é necessário conhecer qual o valor pago pela sua hora de aula e a área que atua. Para os aprendizes, é necessário conhecer a sua formação profissional e a ocupação (cargo que ocupa). Um tutor pode ser um aprendiz e vice-versa. Existem outros tipos de usuários no sistema que não são tutores nem aprendizes.
- Os cursos podem estar associados a nenhum ou vários aprendizes, mas devem estar associados a um e somente um tutor. Um usuário pode ser tutor de nenhum ou vários cursos. Além disso, alguém também pode ser aprendiz de nenhum ou vários cursos. Contudo, um usuário aprendiz em um curso possui uma nota (nota alcançada no curso), data de início e a data de conclusão do curso. Um curso pode ser realizado por um aprendiz mais uma vez, contanto que não seja na mesma data de início.
- Todo curso tem um código identificador, um nome (valor único), descrição e um tipo (Especialização ou Curta Duração ou Treinamento). Além disso, todo curso tem uma duração recomendada (em meses).
- Cada curso pode estar associado a um ou vários exames. Cada exame tem um título, um objetivo e está associada a um e somente um curso. Além disso, os exames têm um código que o identifica para um determinado curso. Este código pode repetir para cursos diferentes. Quando um curso é removido, os exames também são excluídos. Um aprendiz pode fazer 1 (mínimo) ou vários exames e um exame pode ser feito por nenhum ou vários aprendizes. Contudo ao fazer um exame, um aprendiz recebe uma nota.

- O sistema deve manter informações sobre as atividades contendo o código (identificador), o nome da atividade, descrição, a duração prevista (em dias).
- Uma atividade pode ser realizada por vários aprendizes em vários cursos. Um aprendiz matriculado em um curso deve realizar 1 (mínimo) ou várias atividades, uma atividade em um curso pode ser realizada por 0 ou vários aprendizes. Um aprendiz realiza uma determinada atividade somente em um curso, ou seja, se o aprendiz realiza uma atividade em um curso, não deve realizar a mesma atividade em outro curso.

8) Elabore as consultas abaixo em SQL:

- Retorne o nome e o email de todos os usuários
- Retorne o nome, a formação e a ocupação (cargo) de todos os aprendizes
- Retorne o nome do tutor, o valor da hora aula e o nome do curso para todos os cursos que têm a palavra “computação” no nome.
- Retorne o código de todos os usuários que são aprendizes e não são tutores.
- Retorne o nome e o login de todos os usuários que são aprendizes e tutores
- Retorne o nome dos tutores que não são aprendizes
- Retorne o nome do curso e o nome de todos os aprendizes, incluindo os cursos que não possuem aprendizes
- Retorne as informações do usuário e o cargo que ocupa como aprendiz, incluindo os usuários que não são aprendizes.

9) Considerando as tabelas descritas abaixo, elabore o modelo ER.

- Time(id,nome,descrição)
- Premio(idTime,premio) idTime referencia Time(id)
- Jogador(cod,nome,idTime) idTime referencia Time(id)
- Jogo(idTimeCasa, idTimeVisitante, qtdegolTimeCasa,qtdegolTimeVisitante)
idTimeCasa referencia Time(id) e idTimeVisitante referencia Time(id)

10) Considerando o modelo ER e as tabelas elaboradas no exercício 1, elabore as seguintes consultas em SQL:

- Retornar o nome e o CGCJ de todos os clientes que possuem o número entre 45 e 162
- Retornar os dados de todos os desenvolvedores cujo custo é maior que 5500
- Retornar os dados dos projetos do cliente cujo nome é igual a José da Silva Xavier
- Retornar os dados das alocações (código do desenvolvedor, de e até) do projeto de id = 101
- Retornar o nome do projeto e os dados das alocações (id do desenvolvedor, de e até) do projeto de id = 101
- Retornar os dados das alocações (id do projeto, de e até) dos desenvolvedores cujo custo é maior que 2500
- Retornar o nome do projeto, os dados das alocações (de e até) e o nome dos desenvolvedores de todos os desenvolvedores cujo custo é maior que 2500