

Questão 1: Dependências funcionais

Para a relação $R(A, B, C, D)$ com as seguintes dependências funcionais:

$$\{A\} \rightarrow \{B\}, \{B, C\} \rightarrow \{A\}, \{D\} \rightarrow \{A\}$$

Encontre todas as chaves candidatas de R.

Questão 2 (2018-2): Jogo dos vários erros.

Em um projeto de sistema de informação usando banco de dados relacional o banco foi criado com um *script* SQL.

O tempo passou, o sistema está em produção, e uma alteração na estrutura do banco de dados é necessária.

Gerente: “Estagiário, precisa alterar o banco de dados assim, etc”.

Estagiário: “Pode deixar chefe!”

O estagiário alterou o *script* original de criação do banco de dados e rodou, com sucesso, esse novo script de criação. Nem precisou de senha de *root*: usou a conta e senha da própria aplicação, que estavam ali *committed* no repositório mesmo!

Gerente: “Santas máquinas de Turing! Cadê os dados! Sumiu a base!”

Estagiário: “...” 🙄(ツ)🙄

Ache os erros neste cenário. Para cada erro proponha uma solução. Bonus: como você lidaria com essa situação no papel do gerente?

Questão 3 (2018-2): MySQL tem dois tipos principais de índice: *hash tables* e *B-trees*.

Hash tables (ou *hashmaps*) são tabelas de *look-up* organizadas pelo valor de *hash* da chave.

B-trees são uma generalização da árvore binária de busca. São árvores de busca com um *branching factor* maior que dois, e são mantidas balanceadas graças a seus algoritmos de inserção e remoção de nós.

Em ambos os casos a chave vem da coluna que está sendo indexada, e o valor armazenado na estrutura de busca é a posição do registro procurado na tabela.

a) Em uma aplicação financeira temos uma tabela com o valor do saldo do cliente e outras informações deste (nome, número da conta, RG, etc). Suponha que desejamos rapidamente buscar clientes por faixa de saldo (e.g.

SELECT * FROM clientes WHERE saldo >= @saldo_min and saldo <= @saldo_max).

Seria uma boa ideia fazer um índice na coluna saldo? Se sim, qual o tipo preferido: *hash table* ou *B-tree*? Explique.

b) Nesta mesma aplicação financeira e nesta mesma tabela de informações de cliente temos o telefone principal do cliente. Suponha que desejamos frequentemente buscar clientes pelo número de telefone (e.g.

SELECT * FROM clientes WHERE telefone = @telefone_procurado).

Seria uma boa ideia fazer um índice na coluna telefone? Se sim, qual o tipo preferido: *hash table* ou *B-tree*? Explique.

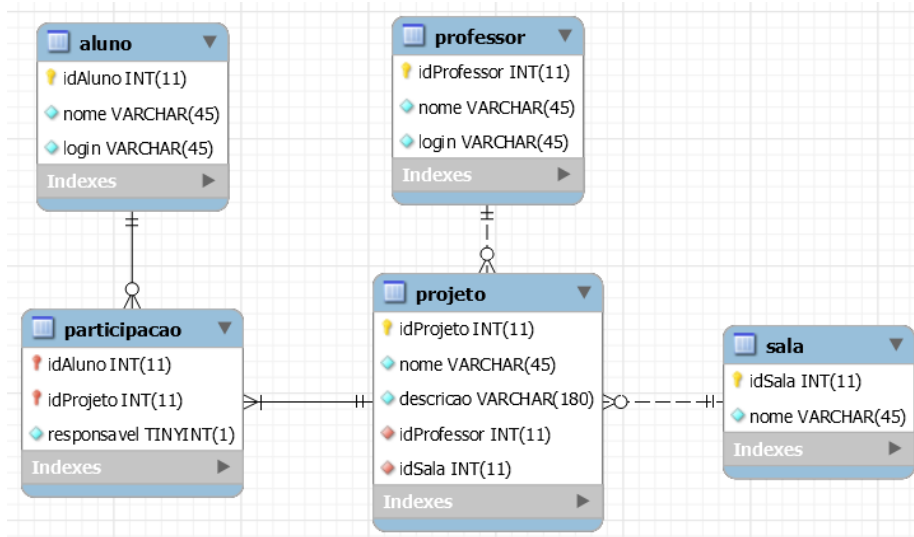
Questão 4 (2018-2):

A Expo Engenharia 2018 vem aí! Será a quarta edição da nossa Expo Engenharia, que está tão grande esse ano que teremos que usar dois andares: o 4º andar e parte do 2º andar (salas 204/205). Vai ser dia 6 de dezembro, uma quinta-feira, das 19h às 22h.

Esse ano vamos precisar de um CRUD para inscrição dos projetos, pois o Google Forms já não estava servindo. Temos os seguintes requisitos:

- O aluno entra no sistema escolhendo seu nome/login em uma lista de alunos. (Isto é uma versão simplificada, na versão real vou também adicionar senha, etc. Ou não – mau professor, mau professor...)
- O aluno cadastra novos projetos. O projeto tem um nome, uma descrição, um professor associado (que conhece o projeto – e.g. o professor da disciplina) e uma sala preferida para exibição (especialmente crítico para os labs).
- O aluno adiciona seus colegas ao projeto. Cada projeto está associado a vários alunos, sendo que pelo menos um deles deve ser declarado responsável pelo projeto.

Eis a primeira versão da modelagem de banco de dados para o projeto:



Dicionário de dados:

Tabela	Campo	Descrição
aluno	nome	Nome do aluno
	login	Login Insper do aluno (para mandar e-mails para <login>@al.insper.edu.br)
professor	nome	Nome do professor
	login	Login Insper do professor (para mandar e-mails para <login>@insper.edu.br)
projeto	nome	Nome do projeto
	descrição	Descrição do projeto
	idProfessor	Professor que conhece o projeto (para levantar questões de segurança)
	idSala	Sala ou laboratório preferencial para exibição do projeto
sala	nome	Nome da sala ou laboratório
participação	idAluno/idProjeto	Conecta aluno com projeto
	responsavel	Booleana que indica que o aluno idAluno é o responsável pelo projeto idProjeto

a) Verdadeiro (V) ou falso (F):

<input type="checkbox"/>	Podemos ter alunos sem projeto
<input type="checkbox"/>	Podemos ter projeto sem sala
<input type="checkbox"/>	O login do professor poderia ser a chave primária da tabela professor
<input type="checkbox"/>	O projeto satisfaz a segunda forma normal se consideramos que o nome é indivisível
<input type="checkbox"/>	A relação entre participação e aluno é uma relação não-identificadora

Nas questões a seguir, escreva código SQL para as tarefas pedidas.

b) Crie a tabela participação

c) Adicione uma coluna para guardar o ano da Expo (pois esse CRUD servirá para os anos seguintes também) e outra coluna para guardar o semestre do projeto (e.g. 1 para Design de Software, 4 para Camada Física da Computação, etc).

d) Liste os alunos que são responsáveis por mais de um projeto.

e) Liste os professores que não estão ligados a nenhum projeto.