

Trabalho de Implementação de BD - 2020.2

Prof. Alexandre Plastino

Neste trabalho, cada grupo (de no máximo quatro alunos) deverá projetar e criar um banco de dados para armazenar as informações e dados necessários ao desenvolvimento de um aplicativo de vacinação (eVacina).

Deverão ser consideradas informações sobre: os diferentes tipos de vacina, os produtores de vacinas, os postos de vacinação, os cidadãos, o histórico de vacinação dos cidadãos, e outros elementos importantes para que o aplicativo possa registrar as vacinas que um cidadão já tomou e para avisá-lo da necessidade de tomar outras vacinas obrigatórias ou recomendáveis, assim como controlar a aplicação das diferentes doses.

O sistema deverá apoiar também o governo a implementar e controlar as campanhas de vacinação, permitindo: o controle dos estoques nos postos de vacinação de cada município, o registro do frasco e do lote da vacina aplicada em cada cidadão, assim como o funcionário do posto que aplicou.

Tentem pensar em funcionalidades interessantes para esse aplicativo e a respectiva necessidade de dados. O grupo que apresentar o melhor trabalho, na avaliação do monitor da disciplina, receberá (cada aluno) 1,0 ponto na nota da P1.

Fase I

Defina o esquema conceitual do banco de dados, utilizando o modelo Entidade-Relacionamento. Adote a notação min-max. Os que quiserem poderão utilizar a ferramenta DIA, disponível em <https://sourceforge.net/projects/dia-installer/>, para editar o diagrama ER.

Fase II

Realize o mapeamento lógico-relacional a partir do esquema conceitual resultante da Fase I. Além do esquema relacional, apresente os comandos SQL para a criação das respectivas tabelas. O banco de dados em questão deverá ser criado no SGBD relacional MySQL. Portanto, confirme a sintaxe dos comandos SQL desse SGBD para a criação das tabelas. Para validar o banco gerado, sugiro pensar em tuplas de exemplo para cada tabela criada de forma a verificar se dados reais conseguem ser armazenados no banco de dados criado.

Fase III

Crie o banco de dados relacional resultante da Fase II no MySQL. Para tanto, solicite ao monitor da disciplina acesso a esse SGBD no laboratório.

Em uma data entre os dias 29/março e 30/abril, vocês deverão, em dupla, apresentar ao monitor Gabriel o banco de dados criado. Os grupos que tiverem três alunos deverão se organizar em dupla e individualmente. Não serão permitidas apresentações realizadas por três alunos.

Para a apresentação do trabalho, cada dupla deverá agendar com o monitor, antecipadamente, um dos horários (40 minutos) indicados na agenda a ser disponibilizada. Não deixe para agendar na última hora, pois o monitor não poderá atender a todos no último dia (os primeiros a agendar poderão escolher data e horário).

Até a véspera da data da apresentação, cada grupo deverá enviar um e-mail para o monitor Gabriel (araujo_gabriel@id.uff.br) contendo: (a) o esquema conceitual (que deverá estar no formato do exemplo fornecido: arquivo Exemplo-ATR.doc) e (b) o esquema relacional (diagrama relacional e comandos SQL) do banco de dados criado. Apenas a primeira dupla de cada grupo precisará enviar este e-mail. Além disso, cada aluno (individualmente) deverá enviar esse material como resposta à atividade criada no Classroom.

Nessa apresentação, a dupla deverá explicar ao monitor o seu esquema conceitual, o mapeamento lógico-relacional realizado e deverá realizar consultas e atualizações no banco de dados criado utilizando os comandos SQL SELECT, INSERT, DELETE e UPDATE. Os seguintes testes deverão ser realizados com o objetivo de mostrar que o SGBD em questão garante a integridade das chaves e a integridade referencial:

(a) Escolha uma chave definida e mostre que o SGBD não admite inserção de valores duplicados. Ou seja, através de um INSERT, mostre que a inserção de um valor duplicado gera um erro.

(b) Escolha três chaves estrangeiras definidas e mostre o funcionamento das opções adotadas (RESTRICT, CASCADE e SET NULL) para a operação de remoção e atualização dos valores referenciados. Por exemplo, através de uma seqüência SELECT, DELETE, SELECT, mostre a propagação da operação de remoção de um valor referenciado por uma chave estrangeira.

(c) Escolha uma chave estrangeira definida e mostre, através de um comando INSERT, que a tentativa de inserção, na chave estrangeira, de um valor não existente na chave primária referenciada, gera um erro de integridade referencial.

Antes de apresentar o trabalho ao monitor, façam testes no MySQL para evitar surpresas indesejáveis na hora da apresentação. Levem para a apresentação os comandos INSERT necessários para o preenchimento da base de dados.

Qualquer dúvida sobre este enunciado: e-mails para plastino@ic.uff.br.

Bom trabalho,
Plastino.

PS: exemplos básicos de utilização dos comandos SQL SELECT, INSERT, DELETE e UPDATE:

SELECT *		
FROM S;	/* Recupera os dados dos fornecedores.	*/
SELECT PNAME		
FROM P		
WHERE COLOR = "Verde";	/* Recupera o nome das peças verdes.	*/
UPDATE P		
SET COLOR = "Azul"		
WHERE P# = "P1";	/* Atualiza a cor da peça P1 para azul.	*/
DELETE		
FROM S		
WHERE S# = "S2"	/* Remove o fornecedor S2.	*/
DELETE		
FROM S;	/* Remove todos os fornecedores.	*/
INSERT		
INTO SP		
VALUES ("S9","P9",200)	/* Insere um novo fornecimento na tabela SP.	*/