

Considere o esquema relacional abaixo:

Amigo (aid: integer, anome: string, sexo: char, idade: integer)

Presente (pid: integer, pdescricao: string, pvalor: real)

ListaPresente (aid: integer, pid: integer, preferência: integer)

AmigoSecreto(aid: integer, aid_amigo: integer, pid_recebido: integer)

1. Questão

Defina um gatilho ("*trigger*") a fim de assegurar que um participante só pode ter no máximo 10 presentes e no mínimo 1 presente em sua lista de presentes.

2. Questão

Defina um gatilho ("*trigger*") a fim de impedir que um participante receba um presente que não está na sua lista de presentes.

3. Questão

Defina um procedimento armazenado ou uma função que recebe como parâmetro de entrada o identificador de um amigo e insere todos os presentes em sua lista de presente.

4. Questão

Escreva um procedimento armazenado ou uma função que recebe como parâmetro de entrada o identificador de um amigo e retorna quanto em dinheiro seria gasto se ele ganhasse todos os presentes de sua lista.

5. Questão

Escreva um procedimento armazenado ou função que recebe como parâmetro de entrada o identificador de um presente e o exclui de qualquer tabela de dependência.

6. Questão

Defina um gatilho ("*trigger*") a fim de impedir que quando um presente for removido, todas as suas dependências sejam removidas.