TRABALHO DE BANCO DE DADOS –1. BIMESTRE

Grupo: Hicham Karim Jebai, Enzo Manente Ferreira.

Em prova será cobrado os conhecimentos adquiridos neste trabalho.

Peso 4,0 – Entregue em pdf, via TEAMS, os diagramas e SQLs utilizados, bem como as respostas para cada exercício.

Demais itens, serão apresentados no dia.

Este trabalho é composto de 3 partes com diferentes pesos. Entretanto, cada parte não realizada/não entregue implicará em -2,0 pontos na nota final do trabalho –limitado a zerar a nota de entrega.

Apresentar em: 09/10/2023–Todos os alunos devem apresentar e estarem preparados para responder quaisquer dúvidas. Notas podem ser diferenciadas.

PARTE 1 – Peso 1,5

1. Dado a modelagem desnormalizada sobre o estudo de caso Matrícula do Aluno discutida em sala de aula, faça:

1. Implemente a tabela num SGBD–SOMENTE A TABELA DESNORMALIZADA.

Para criar a tabela solicitada no SQL foram utilizados os comandos:

CREATE TABLE MatriculaAluno (

id INT AUTO\_INCREMENT PRIMARY KEY,

nomeAluno VARCHAR(255),

cpfAluno VARCHAR(15),

foneAluno VARCHAR(15),

emailAluno VARCHAR(255),

dtMatriculaAluno DATE,

nomeCurso VARCHAR(255),

anoLetivo INT,

nomeDisciplina VARCHAR(255),

nomeProfessorDisciplina VARCHAR(255),

emailProfessor VARCHAR(255)

);

Texto

Descrição gerada automaticamente

A imagem representa a tabela no banco de dados após a sua criação. O campo ID foi definido como primary key, além de ser incrementado automaticamente pelo SGBD.

Os inserts no banco de dados foram realizados por meio de comandos via prompt para esta abordagem;

insert into matriculaaluno (nomeAluno, cpfAluno, foneAluno, emailAluno, dtmatriculaAluno, nomeCurso, anoLetivo, nomeDisciplina, nomeProfessorDisciplina, emailProfessor) Values('JoseABC de D', '111.222.333-44', '99999-9999', 'joseabc@xx.com', '2023-10-05', 'ciencia da computacao', 2023, 'Banco de Dados', 'Roberto Gil', 'emailprof@xx.com');

após a realização dos inserts e com a tabela já preenchida com os dados que seriam necessários temos:

Tela de computador

Descrição gerada automaticamente

1. Crie um programa em JAVA para conexão com o SGBD e realize os cenários descritos nos próximos itens –não altere a sequência de execução;

Temos a conexão do SGBD com o programa em java, para realizar os próximos cenários a solução escolhida foi mostrar os resultados no prompt por meio de System.out.println(“”);

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

1. Considere uma maneira do sistema mostrar ao futuro alunos quais os cursos disponíveis;

Para este item obtivemos a seguinte solução:

Texto

Descrição gerada automaticamente

1. Considere uma maneira do sistema mostrar ao futuro aluno quais as disciplinas disponíveis para um determinado curso;

Para este item obtivemos a seguinte solução:

Texto

Descrição gerada automaticamente

1. Considere uma maneira do sistema mostrar quais os alunos já cadastrados, porém ainda não matriculados em nenhum curso;

Para este item obtivemos a seguinte solução:

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

1. Matrícula 4 alunos diferentes, e para cada aluno, considere de três a quatro disciplinas por aluno;

Após a inserção dos alunos na tabela temos:

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

1. Nos quatro alunos, considere que pelo menos dois deles estarão sendo matriculados na disciplina de Banco de Dados. Porém no nome da disciplina, para um insira como “Banco de Dados” e para o outro “BD”;

Este item foi realizado por meio de inserts no prompt do SGBD, os comandos utilizados foram:

insert into matriculaaluno (nomeAluno, cpfAluno, foneAluno, emailAluno, dtmatriculaAluno, nomeCurso, anoLetivo, nomeDisciplina, nomeProfessorDisciplina, emailProfessor) Values('JoseABC de D', '111.222.333-44', '99999-9999', 'joseabc@xx.com', '2023-10-05', 'ciencia da computacao', 2023, 'Banco de Dados', 'Roberto Gil', 'emailprof@xx.com');

insert into matriculaaluno (nomeAluno, cpfAluno, foneAluno, emailAluno, dtmatriculaAluno, nomeCurso, anoLetivo, nomeDisciplina, nomeProfessorDisciplina, emailProfessor) Values('Hicham', '222.333.444-55', '99999-8888', 'hicham@xx.com', '2023-10-05', 'ciencia da computacao', 2023, 'BD', 'Roberto Gil', 'emailprof@xx.com');

1. Um dos alunos deve se chamar “José ABC de D”;

Este item foi realizado através de um insert no prompt do SGBD:

insert into matriculaaluno (nomeAluno, cpfAluno, foneAluno, emailAluno, dtmatriculaAluno, nomeCurso, anoLetivo, nomeDisciplina, nomeProfessorDisciplina, emailProfessor) Values('JoseABC de D', '111.222.333-44', '99999-9999', 'joseabc@xx.com', '2023-10-05', 'ciencia da computacao', 2023, 'Banco de Dados', 'Roberto Gil', 'emailprof@xx.com');

Aqui estão mais algumas disciplinas cursadas por “JoseABC de D”:



1. . Implemente uma interface qualquer em java (pode ser apenas utilizando System.out.println(“”) para mostrar uma matrícula do aluno realizada;

Interface utilizando System.out.println(“”); para mostrar os dados de uma matrícula realizada do aluno:

Texto

Descrição gerada automaticamente

Resultados da pesquisa para o aluno: Enzo:

Texto

Descrição gerada automaticamente com confiança média

1. Liste todos os alunos que estão matriculados na disciplina de Banco de Dados. Alguém não apareceu? Por quê?

Ao listar todos os alunos matriculados em Bando de dados, temos:

Texto

Descrição gerada automaticamente

O aluno Hicham não aparece na pesquisa da disciplina de banco de dados, pois no SGBD o nome da disciplina na matrícula foi feito como ‘BD’, quando realizamos a busca apenas as matriculas com “Banco de Dados” são exibidas.

PARTE 2 –Peso 0,5

A partir do estudo de caso normalizado do trabalho 1, considere em seu cadastro, um aluno matriculado em 2015 na disciplina de Banco de Dados, onde o professor foi o “João ABCD” e outro aluno matriculado em 2023 onde o professor foi o “Roberto Gil”.

Baseado nos dados acima, implemente o relacionamento entre disciplina e professor comas seguintes variantes:

a) Proposta de Solução 1:

Destacamos aqui que o modelo proposto permite que uma disciplina tenha apenas um professor, uma vez que uma chave estrangeira para o professor é mantida na tabela da disciplina.

Implementação em SQL dessa proposta:

CREATE TABLE Professor (

codProfessor INT PRIMARY KEY,

nomeProfessor VARCHAR(255)

);

CREATE TABLE Disciplina (

codDisciplina INT PRIMARY KEY,

nomeDisciplina VARCHAR(255),

codProfessor INT,

FOREIGN KEY (codProfessor) REFERENCES Professor(codProfessor)

);

Impacto:

Esse modelo é simples, mas limita a flexibilidade para expressar as relações entre as entidades. Ele não suporta o conceito de uma disciplina sendo lecionada por vários professores diferentes em anos letivos diferentes.

1. Proposta de Solução 2

Este modelo propõe a utilização de uma tabela de junção (DisciplinaProfessor) para gerenciar as múltiplas relações entre professores e disciplinas. Isso permite uma flexibilidade muito maior, em que uma disciplina pode ser lecionada por diferentes professores em diferentes anos letivos.

Implementação em SQL dessa proposta:

CREATE TABLE Professor (

codProfessor INT PRIMARY KEY,

nomeProfessor VARCHAR(255)

);

CREATE TABLE Disciplina (

codDisciplina INT PRIMARY KEY,

nomeDisciplina VARCHAR(255)

);

CREATE TABLE DisciplinaProfessor (

codDisciplina INT,

codProfessor INT,

anoLetivo INT,

dtinicioNaDisciplina DATE,

PRIMARY KEY(codDisciplina, codProfessor, anoLetivo, dtinicioNaDisciplina),

FOREIGN KEY (codDisciplina) REFERENCES Disciplina(codDisciplina),

FOREIGN KEY (codProfessor) REFERENCES Professor(codProfessor)

);

Impacto:

Este modelo permite uma grande flexibilidade e acomoda muitos casos do mundo real, pois uma disciplina pode ser lecionada por diferentes professores em diferentes anos letivos. No entanto, o gerenciamento dos dados é um pouco mais complexo e requer mais consultas inter-tabelas, o que pode impactar no desempenho se as tabelas se tornarem muito grandes.

PARTE 3 –Peso 2,0

Considerando a modelagem na 3.FN já apresentado em sala de aula para matricula de Aluno, realize os seguintes ajustes:

1. Considerar Endereço completo do Aluno e professor

Para o endereço de aluno e professor temos: as tabelas

CREATE TABLE Endereço (

idEndereço INT AUTO\_INCREMENT PRIMARY KEY,

Cep VARCHAR(8),

idProfessor INT,

idBairro INT,

idcidade INT,

FOREIGN KEY (idProfessor) REFERENCES Professor(idProfessor),

FOREIGN KEY (idBairro) REFERENCES Bairro (idBairro),

FOREIGN KEY (idCidade) REFERENCES Cidade(idCidade)

);

CREATE TABLE Bairro(

idBairro INT AUTO\_INCREMENT PRIMARY KEY,

nomeBairro VARCHAR(45) NULL

);

CREATE TABLE Cidade (

idCidade INT AUTO\_INCREMENT PRIMARY KEY,

nomeCidade VARCHAR(45) NULL

);

1. Tanto para aluno, quanto para professor, registrar seus fones e e-mails

Para registrar fone de Aluno e professor temos:

CREATE TABLE Professor (

idProfessor INT AUTO\_INCREMENT PRIMARY KEY,

nomeProfessor VARCHAR(45),

emailProfessor VARCHAR(45),

foneProfessor VARCHAR(45));

CREATE TABLE Aluno(

idAluno INT AUTO\_INCREMENT PRIMARY KEY,

nomeAluno VARCHAR(45),

cpfAluno VARCHAR(45),

foneAluno VARCHAR(45),

emailAluno VARCHAR(45),

idEndereço INT,

FOREIGN KEY (idEndereço) REFERENCES Endereço(idEndereço));

c)No momento da matrícula no Ano Letivo o sistema deve validar:

•Se o aluno já está matriculado no Curso informado;

•Se o Ano Letivo é válido. Por exemplo, querer matricular no ano letivo de 2024.

Para a validação do ano letivo, foi implementado no SGBD:

DELIMITER //

CREATE TRIGGER valida\_matricula

BEFORE INSERT ON MatriculaAluno

FOR EACH ROW

BEGIN

DECLARE num\_matriculas INT;

-- Verificar se o aluno já está matriculado no mesmo curso e ano letivo

SELECT COUNT(\*) INTO num\_matriculas

FROM MatriculaAluno

WHERE idAluno = NEW.idAluno

AND idCurso = NEW.idCurso

AND YEAR(anoLetivo) = YEAR(NEW.anoLetivo);

IF num\_matriculas > 0 THEN

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Aluno já está matriculado no mesmo curso no ano letivo informado';

END IF;

END;

//

DELIMITER ;

1. Apresente via PDF o modelo E-R:

Se encontra no Arquivo MODELO\_E-R;

1. Implemente todas as tabelas e escreva um programa que se conecte ao SGBD, escrevendo lógicas e consultas que atendam as questões abaixo:

Para este item foram criada as seguintes tabelas:

Texto

Descrição gerada automaticamente

1. Considere uma maneira do sistema mostrar ao futuro alunos quais os cursos disponíveis

Função e comando para pegar todos os cursos disponíveis e imprimir:

Texto

Descrição gerada automaticamente

1. Considere uma maneira do sistema mostrar ao futuro aluno quais as disciplinas disponíveis para um determinado curso

Função e comandos para pegar e imprimir todos as disciplinas de um curso;

Texto

Descrição gerada automaticamente

1. Considere uma maneira do sistema mostrar quais os alunos já cadastrados, porém ainda não matriculados em nenhum curso

Tela de computador

Descrição gerada automaticamente

1. Matrícula 4 alunos em dois cursos diferentes

j) Liste quais alunos estão num curso e em outro

k) Para cada aluno matriculado num curso, matricule os no ano letivo de 2023, sendo que cada aluno esteja em algumas disciplinas

l) Liste os alunos matriculados no curso de Ciência da Computação que fazem a disciplina de Banco de Dados em relação ao realizado na PARTE1:

m) O que se observou de diferente desta vez, do ponto de vista da implementação (Java / SQL)?

n) O que você observou de diferente entre os dois modelos de dados, considerando as possíveis funcionalidades e requisitos do futuro software?