Trabalho para as férias

Prof. Ana Rúbia Martins

Alunos: Karen Bastos e Arthur Alves Soares

Turma 241A NOITE - ES

# Banco de Dados

**Questão 1)** Deﬁna o que é um Banco de Dados. Explique sua importância e alguns exemplos de aplicação.

O software de banco de dados seria um local onde todos as informações ficam armazenadas e permite a gestão destes dados. É usado para criar, editar e manter arquivos e registro de dados, facilitando o armazenamento de entradas, edição e atualização de relatórios em dados.

**Questão 2)** Explique a diferença entre um Banco de Dados Relacional e um Banco de Dados Não-Relacional.

O Banco de Dados Relacional armazenam os dados em tabelas com colunas e linhas, e estabelece relaçoes entre elas atraves de chaves estrageiras, isso permite que is dados sejam acessados e consultados atraves de consultas SQL.

Já o Banco de Dados Não-Relacional armazenam os dados em documentos, que são coleções de campos com valores, conhecido como NoSQL é mais adequado para lidar com grandes quantidades de dados estruturados e nao estruturados.

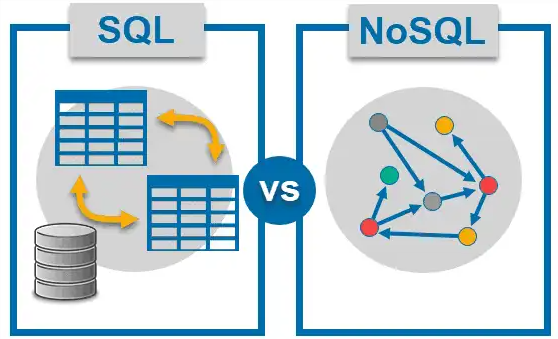


Imagem de exemplo dos dois tipos de banco relacionados e não relacionados

**Questão 3)** O que é um Sistema de Gerenciamento de Banco de Dados (SGBD)? Qual a sua função principal?

É um conjunto de softwares que permite o gerenciamento da bases de dados, responsaveis por acessar, controlar, organizar e proteger as informações de uma aplicação, seu princial objetivo é gerenciar as bases de dados utilizadas.

**Questão 4)** Descreva os principais componentes de um banco de dados relacional. Explique o que são tabelas, registros, e campos.

As tabelas são estruturas que armazenam dados em formato tabular.

Os registros são representado pelo significado de tuplas (ou registros), em que cada linha é composta por uma série de campos que contém um valor de atributo identificado por um nome.

Os campos conhecido tambem como atributo(colunas) que pode representar as características de um registro da tabela, representando um atributo do dado.

Valor do campo

tb\_usuarios

Nome do campo

|  |  |  |  |
| --- | --- | --- | --- |
| id | nome | cpf | idade |
| 1 | Karen Bastos | 12345678901 | 29 |
| 2 | Arthur Alves | 12345678902 | 18 |
| 3 | Mel Maia | 12345678903 | 25 |

Registros/Linhas/Tuplas

Campos/Colunas/Atributos

**Questão 5)** O que é uma chave primária e uma chave estrangeira? Qual a importância delas no contexto de bancos de dados relacionais?

Chave Primária: Identificador único para cada registro em uma tabela. Exemplo: id em uma tabela de usuários.

Chave Estrangeira: Campo que cria uma ligação entre duas tabelas, referenciando a chave primária de outra tabela.

tb\_usuarios

PK

|  |  |  |  |
| --- | --- | --- | --- |
| id | nome | cpf | idade |
| 1 | Karen Bastos | 12345678901 | 29 |
| 2 | Arthur Alves | 12345678902 | 18 |
| 3 | Mel Alves | 12345678903 | 25 |

Interligando as tabelas

Chave primária da tb\_usuarios

Chave estrangeira da tb\_responsaveis

|  |  |  |
| --- | --- | --- |
| id\_resp | nome  tb\_responsaveis | id\_usuario  FK |
| 10 | Maria Alves | 2 |
| 20 | Pedro Bastos | 1 |
| 30 | Julia Alves | 3 |

**Questão 6)** Explique os conceitos de normalização e desnormalização em bancos de dados. Qual é a importância da normalização?

Normalização é o processo de otimização da estrutura dos dados para evitar redundâncias e inconsistências, enquanto a desnormalização é o processo de adicionar redundâncias propositalmente para melhorar o desempenho de consultas frequentes.

A importância da normalização é um processo de adequar o empreendimento as normas técnicas exigidas, para a proteção dos dados e tornar o banco de dados mas flexivel, eliminando as redundâncias e inconsistências.

**Questão 7)** Descreva o que é uma transação em um banco de dados e explique as propriedades ACID.

Transação em um banco de dados é qualquer função executada em um banco de dados com uma unidade lógica de trabalho, essa função tem dois estados executada ou não executada.

Propriedades do ACID: Atomicidade, Consistência, Isolamento, Durabilidade

**Questão 8)** O que é uma consulta SQL? Forneça exemplos de comandos básicos SQL (SELECT, INSERT, UPDATE, DELETE).

Consulta SQl seria uma linguagem de consulta estruturada para manilucação lógica, com objetivo de facilitar a conversa com o banco de dados a fim de manuziar as informações desejadas. As consultas SQL podem ser executadas de várias formas, segue algumas exemplos:

SELECT – SELEÇÃO DE DADOS

SELECT \* FROM tb\_usuarios WHERE nome LIKE ‘%Alves’;

INSERT – INSERÇÃO DE DADOS

INSERT INTO tb\_usuarios (id, nome, cpf, idade) VALURES (4, ‘Vitória’, ‘12345678904’; 19);

UPDATE – ATUALIZAÇÃO DE DADOS

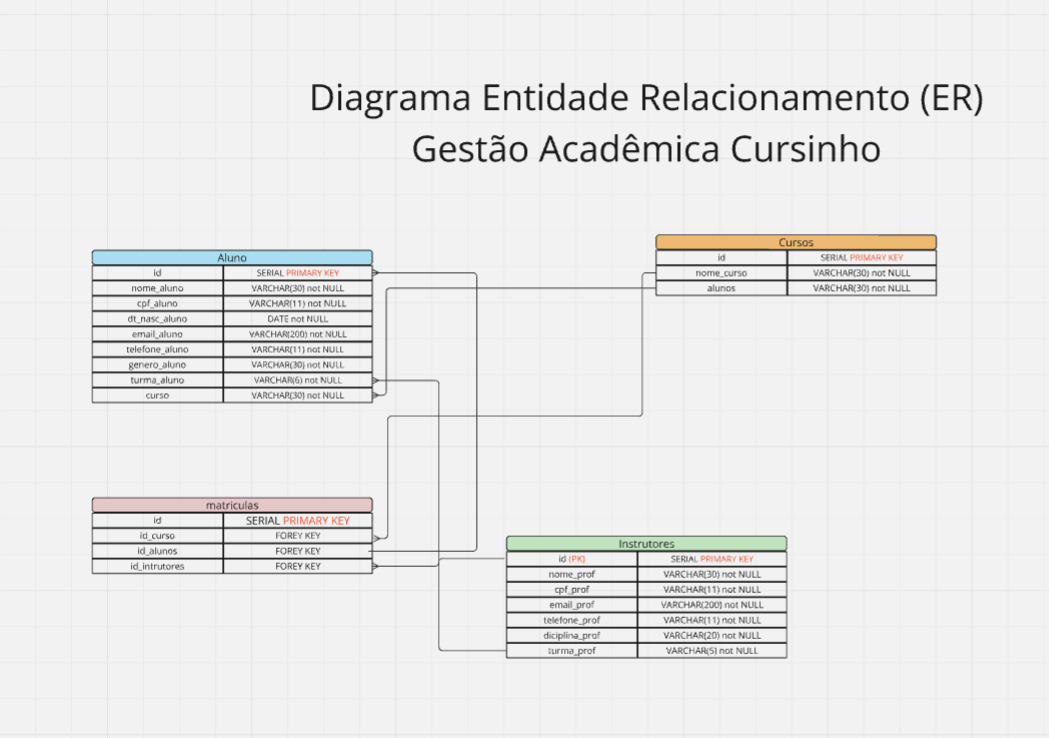
UPDATE tb\_usuarios SET nome = ‘Vitória Afonso’ WHERE (id = '4');

DELETE – EXCLUSÃO DE DADOS

DELETE FROM tb\_usuarios WHERE (id = '4');

# PRÁTICA

1. Crie um modelo ER (Entidade-Relacionamento) para um sistema de gerenciamento de cursos online. Considere as seguintes entidades: Cursos, Instrutores, Alunos, Matrículas.
2. Deﬁna as relações entre essas entidades e identiﬁque as chaves primárias e estrangeira



N,1

N,N

1,1

1,N

1,1

**Python**

**Questão 1)** Explique o que é Python e quais são suas principais características.

Python é uma linguagem de programação versátil que pode ser usada para criação de projetos simplas até projetos ccomplexos, conhecido por sua simplicidade e legibilidade e facil integração com outras tecnologias web comuns.

**Questão 2)** Descreva os tipos de dados básicos em Python e forneça exemplos de cada um.

Caracter = str (sequencia de caracteres como o nome de uma pessoa)

Real = float (representa números reais, com casas decimais = 2,3339)

Inteiro = int (representa números inteiros, positivos ou negativos, sem casas decimais = 2)

Lógico = bool (Verdadeiro (True) ou Falso (False))

**Questão 3)** O que são listas e dicionários em Python? Compare esses dois tipos de estruturas de dados.

Dicionários – são uma coleção que guarda varios valores muldimencionais para cada indice. Usamos um par de chaves para denotar umdicionario. Exemplo: meu\_dict = {'nome': 'Alice', 'idade': 25}

Lista – Lista encadeada, que guarda apenas um valor por vez. Usamos um par de colchetes para denotar uma lista. Exemplo: ['a', 'b', 'c']

**Questão 4)** Explique o conceito de funções em Python.

Função é um bloco de código que realiza uma tarefa específica, utilizamos a palavra def segundo pelo nome da função e parenteses, para definir a função. As funções podem reutilizar códigos promovendo organização e legibilidade deixando de mameira mais organizada os códigos.

Exemplo:

def funcoes():

Print(“Funções do Python”)

funcoes()

# Prática

1. Escreva uma função que leia as palavras em words.txt e guarde-as como chaves em um

[dicionário](https://pense-python.caravela.club/11-dicionarios/09-glossario.html#dicion%C3%A1rio). Não importa quais são os valores. Então você pode usar o operador in como uma forma rápida de veriﬁcar se uma string está no dicionário.

lista = []

while True:

print("1) Adicionar contato")

print("2) Remover contato")

print("3) Exibir Agenda")

print("4) Sair")

opcao = int(input("Digite uma opção: "))

if opcao == 0:

break

elif opcao == 1:

nome = input("Digite o nome do contato: ")

telefone = input("Digite o telefone do contato: ")

contato = {"nome": nome, "telefone": telefone}

lista.append(contato)

print(f" {nome} foi adicionado.")

elif opcao == 2:

nome = input("Digite o nome do contato que deseja remover: ")

contato\_encontrato = False

for contato in lista:

if contato ["nome"] == nome:

lista.remove(contato)

print(f"{nome} foi removido.")

contato\_encontrado = True

break

if contato\_encontrado == False:

print(f"{nome} não foi encontrado.")

elif opcao == 3:

print("\nLISTA DE CONTATOS: ")

for contato in lista:

print(f"Nome: {contato['nome']}")

print(f"Telefone: {contato['telefone']}")

else:

print("Opção inválida!")

1. Faça um pedra, papel, tesoura

pedra = 0

papel = 1

tesoura = 2

jogar\_novamente = "Sim"

while (jogar\_novamente == "Sim"):

jogador1 = int(input("Jogador 1, dígite 0 p/pedra, 1 p/papel ou 2/tesoura: "))

jogador2 = int(input("Jogador 2, dígite 0 p/pedra, 1 p/papel ou 2/tesoura: "))

if (0 <= jogador1 <= 2) and (0 <= jogador2 <= 2):

if (jogador1 == jogador2): #mesma escolha = Empate

print("Empate! Ninguém ganhou.")

elif (jogador1 - jogador2 == -2) or (jogador1 - jogador2 == 1):

print("Jogador 1, você ganhou!")

else:

print("Jogador 2, você ganhou!")

else:

print("Opção inválida.")

jogar\_novamente = input("Você quer tentar novamente? Digite Sim ou Não: " )

print("Até a próxima!" )

1. Duas palavras são anagramas se você puder soletrar uma rearranjando as letras da outra. Escreva uma função chamada is\_anagram que tome duas strings e retorne True se forem anagramas.

#Função

def is\_anagrama(string1, string2):

flag = False

string1\_ordenada = sorted(string1)

string2\_ordenada = sorted(string2)

print(string1\_ordenada == string2\_ordenada)

if(string1\_ordenada == string2\_ordenada):

flag = True

return flag

#verificar

print(is\_anagrama("pato", "topa")) #Exemplo True

print(" ")

print(is\_anagrama("amor", "roma")) #Exemplo True

print(" ")

print(is\_anagrama("bola", "lobo")) #Exemplo False

1. Crie um dicionário em Python para armazenar informações de contatos. O dicionário deve usar os nomes como chaves e os números de telefone como valores. Escreva funções para:

Adicionar um novo contato. Remover um contato existente. Buscar um contato pelo nome. Listar todos os contatos.

lista = []

while True:

print("1) Adicionar contato")

print("2) Remover contato")

print("3) Exibir Agenda")

print("4) Sair")

opcao = int(input("Digite uma opção: "))

if opcao == 0:

break

elif opcao == 1:

nome = input("Digite o nome do contato: ")

telefone = input("Digite o telefone do contato: ")

contato = {"nome": nome, "telefone": telefone}

lista.append(contato)

print(f" {nome} foi adicionado.")

elif opcao == 2:

nome = input("Digite o nome do contato que deseja remover: ")

contato\_encontrato = False

for contato in lista:

if contato ["nome"] == nome:

lista.remove(contato)

print(f"{nome} foi removido.")

contato\_encontrado = True

break

if contato\_encontrado == False:

print(f"{nome} não foi encontrado.")

elif opcao == 3:

print("\nLISTA DE CONTATOS: ")

for contato in lista:

print(f"Nome: {contato['nome']}")

print(f"Telefone: {contato['telefone']}")

else:

print("Opção inválida!")