B - CONHECIMENTO DE COMANDOS SQL.

1. Criar um comando SELECT que retorne APENAS o nome dos clientes da cidade “TUPA”. Utilizar o EXISTS para realizar a condição.

SELECT NOME

FROM CLIENTES

WHERE EXISTS (

SELECT 1

FROM CIDADES

WHERE CIDADES.CODIGO\_CIDADE = CLIENTES.Codigo\_Cidade

AND CIDADES.NOME = 'TUPA'

);

1. Criar um comando SELECT que retorne o nome do cliente e o nome da cidade de todos os registros

SELECT

CLIENTES.NOME AS NOME\_CLIENTE

,CIDADES.NOME AS NOME\_CIDADE

FROM CLIENTES

INNER JOIN CIDADES ON CIDADES.CODIGO\_CIDADE = CLIENTES.CODIGO\_CIDADE

1. Criar um comando SELECT que retorne TODOS os nomes e os códigos dos clientes ordenados por nome do cliente.

SELECT

CLIENTES.CODIGO\_CLIENTE

,CLIENTES.NOME

FROM CLIENTES

ORDER BY NOME

1. Criar um comando DELETE que exclua clientes com código do cliente entre a numeração 100 e 200.

--PRIMEIRO FAÇO UM SELECT NA TABELA TODA PARA IDENTIFICAR O CONTEÚDO

SELECT \*

FROM CLIENTES

--INICIO UMA TRANSAÇÃO PARA PODER CONFIRMAR AS ALTERAÇÕES QUE SERÃO REALIZADAS

BEGIN TRANSACTION;

DELETE

FROM CLIENTES

WHERE CODIGO\_CLIENTE BETWEEN 100 AND 200;

--APÓS O COMANDO DE DELETE, CONFIRMO PARA SABER SE A DELEÇÃO OCORREU CONFORME ESPERADO

SELECT \*

FROM CLIENTES

--AQUI JÁ DEIXO OS COMANDOS DE COMMIT (CASO TENHA OCORRIDO COMO O PLANEJADO A DELEÇÃO)

--OU ROLLBACK, CASO TENHA OCORRIDO ALGO INESPERADO

--COMMIT;

--ROLLBACK;

1. Criar um comando UPDATE que altere o estado de todas as cidades para o estado “SP” quando estiverem com estado igual a “PR”.

--PRIMEIRO FAÇO UM SELECT NA TABELA TODA PARA IDENTIFICAR O CONTEÚDO

SELECT \*

FROM CIDADES

--INICIO UMA TRANSAÇÃO PARA PODER CONFIRMAR AS ALTERAÇÕES QUE SERÃO REALIZADAS

BEGIN TRANSACTION;

UPDATE CIDADES

SET

ESTADO = 'SP'

WHERE ESTADO = 'PR'

--APÓS O COMANDO DE UPDATE, CONFIRMO PARA SABER SE A ATUALIZAÇÃO OCORREU CONFORME ESPERADO

SELECT \*

FROM CIDADES

--AQUI JÁ DEIXO OS COMANDOS DE COMMIT (CASO TENHA OCORRIDO COMO O PLANEJADO A ATUALIZAÇÃO)

--OU ROLLBACK, CASO TENHA OCORRIDO ALGO INESPERADO

--COMMIT;

1. --ROLLBACK;Criar um comando INSERT na tabela de clientes de um registro qualquer com todos os campos da tabela. (valores livres)

INSERT INTO CLIENTES

(CGC\_CPF\_Cliente

,Nome

,Telefone

,Endereco

,Bairro

,Complemento

,E\_mail

,Codigo\_Cidade

,Cep)

VALUES

('13028564592'

,'NATAN ROBERTO XAVIER'

,'11977000061'

,'RUA CLOVIS DE OLIVEIRA'

,'JARDIM ITAIPU'

,'CASA'

,'NATANROBERTO182@GMAIL.COM'

,7

,17602480)

C - LÓGICA DE PROGRAMAÇÃO

1. Monte um arquivo texto com um algoritmo, passo a passo, para troca de um pneu de carro que furou

1. Pare o carro em um local seguro.
2. Puxe o freio de mão do carro.
3. Localizar o triângulo para sinalização da via.
4. Posicionar o triângulo de forma visível aos usuários da via.
5. Localizar o estepe.
6. Pegar o estepe.
7. Localizar o macaco.
8. Pegar o macaco.
9. Localizar a chave de roda.
10. Pegar a chave de roda.
11. Posicionar o macaco para elevar o lado do pneu furado.
12. Com todas as rodas no chão, utilizar a chave de roda para afrouxar os parafusos da roda.
13. Subir o macaco até tirar completamente a roda com o pneu furado do chão.
14. Remover todos os parafusos da roda que está o pneu furado.
15. Retirar a roda com pneu furado.
16. Posicionar o estepe no local que estava o a roda com pneu furado.
17. Colocar os parafusos de forma a prender o estepe no cubo de roda.
18. Abaixar o macaco até todos os pneus estarem no chão.
19. Retirar o macaco.
20. Com auxílio da chave de rodas, apertar os parafusos do estepe.
21. Guardar a roda com pneu furado no compartimento do estepe.
22. Guardar o macaco no compartimento do macaco.
23. Guardar a chave de roda no compartimento da chave de roda.
24. Recolher o triângulo.
25. Guardar o triângulo no compartimento do triângulo.