-- Questão 1 --

-- selecionando o nome e o telefone de todos os clientes contidos na tabela cliente

SELECT Nome, Telefone

FROM tb\_cliente

GROUP BY Nome, Telefone;

-- Questão 2 --

SELECT c.ref\_agente\_de\_viagens.Salario, c.ref\_agente\_de\_viagens.ref\_supervisiona.Nome, c.Valor\_comissao -- para saber sobre a comissao dos agentes, pego a referencia dos agentes contido na tabela c, e para saber o supervisor pego também a referencia de quem o supervisiona

FROM tb\_agente\_de\_viagens a, tb\_comissao c

-- Questão 3 --

SELECT \*

FROM ( SELECT ag.Nome --exponho o nome dos meliantes que tem a maior qtd de telefones

FROM tb\_agente\_de\_viagens a

HAVING COUNT(\*) =

(SELECT MAX(ag.tel) --seleciono quem tem mais telefones

FROM (SELECT COUNT(t1.ddd) as tel --seleciono a quantidade de quem tem mais telefones

FROM tb\_agente\_de\_viagens av, TABLE(av.Telefone) t1) ag)

UNION -- uno as duas tabelas que conterao as mesmas colunas

SELECT cli.Nome

FROM tb\_cliente c

HAVING COUNT(\*) =

(SELECT MAX(cli.tel) --idem só que pra cliente

FROM (SELECT COUNT(t2.ddd) as tel

FROM tb\_cliente cl, TABLE(cl.Telefone) t2) cli)

GROUP BY Nome)

-- Questão 4 --

SELECT DISTINCT Nome, Milhas -- retorna o numero de milhas andadas por todos os clientes

FROM tb\_cliente

-- Questão 5

ALTER TYPE tp\_cliente MEMBER FUNCTION passagens (nomeCliente VARCHAR2) RETURN VARCHAR2 CASCADE;

CREATE OR REPLACE TYPE BODY tp\_cliente AS MEMBER FUNCTION passagens (nomeCliente VARCHAR2) RETURN VARVHAR 2 IS

qtdPass :=0 NUMBER%TYPE;

nomeAgente tb\_agente\_de\_viagens%TYPE;

BEGIN

SELECT a.nome INTO nomeAgente FROM tb\_agente\_de\_viagens

WHERE a.CPF = (SELECT DISTINCT(DEREF(p. --me atrapalhei aki pq nao tem nada pra referenciar a passagem em relaçao ao cliente :/

RETURN nomeAgente;

END passagens;

END;

/

-- Questão 6 --

SELECT MAX(Valor\_comissao) --retorna a maior comissao ganha por um agente

FROM tb\_comissao

-- Questão 7 --

ALTER TYPE tp\_passagem ADD MEMBER PROCEDURE passagemPromo CASCADE

CREATE OR REPLACE TYPE BODY tp\_passagem AS MEMBER PROCEDURE passagemPromo IS

v\_valor tb\_passagem.Valor\_passagem%TYPE; --variaveis

v\_desc tb\_promocao.Desconto%TYPE;

CURSOR novoValor IS

SELECT tp.ref\_passagem.Valor\_passagem --o cursor q passara pela tabela tem\_promocao

FROM tb\_tem\_promocao tp;

BEGIN

OPEN novoValor;

LOOP

FETCH novoValor INTO v\_valor, v\_desc;

EXIT WHEN novoValor%NOTFOUND;

UPDATE v\_valor --aki altero o valor original da passagem aplicando o desconto em %

SET v\_valor = (v\_valor\*v\_desc)/100;

END LOOP;

CLOSE novoValor.

END passagemPromo;

END;

/

-- Questão 8

ALTER TYPE tp\_passagem ADD MEMBER FUNCTION --nao deu pra fazer

-- Questão 9 --

SELECT AVG(Salario) AS mediaSalarial --retorna a media dos salarios dos agentes

FROM tb\_agente\_de\_viagens

-- Questão 10 --

SELECT COUNT(Codigo\_passagem) --retorna o total de passagens vendidas

FROM tb\_passagens

-- legenda (-- questao x --)completa

-- (-- questao x )incompleta