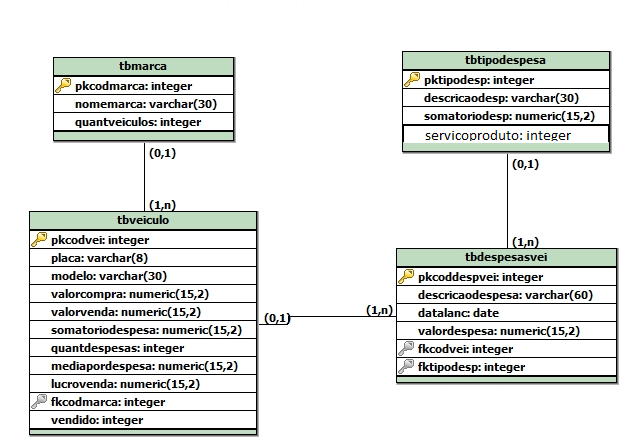
**Trabalho banco II**

**Procedimento, Função e PL/SQL**

Tendo como base o Diagrama ER que segue resolva as questões solicitadas:



1. Crie uma constraint que não permita armazenar no atributo “Valorvenda” da tabela de veículo um valor inferior a zero.

**ALTER TABLE** TBVEICULO **ADD CONSTRAINT** valor\_inferior **CHECK** (VALORVENDA >= 0);

1. Faça uma função que passando **somente o nome da tabela** retorne como resposta o próximo código a ser cadastrado. O próximo código sempre será o maior código incrementado mais um. Por exemplo, se passado como parâmetro “tbveiculo”, a função deverá encontrar o maior código de veículo cadastrado (pkcodvei) e incrementar mais um como resposta. Já se for informado no parâmetro “tbmarca”, a função deverá encontrar o maior código de marca (pkcodmarca) e incrementar mais um como resposta.

**CREATE OR REPLACE FUNCTION** *get\_prox\_cod\_cad*(  
 nome\_tabela **IN VARCHAR2**)  
**RETURN INTEGER  
IS** qtd **INTEGER**;  
**BEGIN  
 IF** (*UPPER*(nome\_tabela) = 'TBMARCA') **THEN  
 SELECT** (*NVL*(*MAX*(PKCODMARCA), -1) + 1) **INTO** qtd **FROM** TBMARCA;  
 **ELSIF** (*UPPER*(nome\_tabela) = 'TBTIPODESPESA') **THEN  
 SELECT** (*NVL*(*MAX*(PKTIPODESP), -1) + 1) **INTO** qtd **FROM** TBTIPODESPESA;  
 **ELSIF** (*UPPER*(nome\_tabela) = 'TBVEICULO') **THEN  
 SELECT** (*NVL*(*MAX*(PKCODVEI), -1) + 1) **INTO** qtd **FROM** TBVEICULO;  
 **ELSIF** (*UPPER*(nome\_tabela) = 'TBDESPESASVEI') **THEN  
 SELECT** (*NVL*(*MAX*(PKCODDESPVEI), -1) + 1) **INTO** qtd **FROM** TBDESPESASVEI;  
 **END IF**;  
  
 **RETURN** qtd;  
**END** *get\_prox\_cod\_cad*;

1. Crie um procedimento que receba **somente o nome da marca** e o procedimento deve cadastrar uma nova marca, chamando a função definida na questão 2.

**CREATE OR REPLACE PROCEDURE** *cad\_marca*(  
 nome\_marca **IN** TBMARCA.NOMEMARCA%**TYPE**)  
**IS** cod\_insert **INTEGER**;  
**BEGIN  
 SELECT** *GET\_PROX\_COD\_CAD*('TBMARCA') **INTO** cod\_insert **FROM** dual;  
  
 **INSERT INTO** TBMARCA (PKCODMARCA, NOMEMARCA, QUATVEICULOS) **VALUES** (cod\_insert, nome\_marca, 0);  
**END** *cad\_marca*;

1. Crie uma função que passado somente a placa do veículo e um número (1 ou 2), onde 1 representa produto e 2 representa serviço, retorne como resposta, a soma de todas as despesas do veículo (valordespesa) da placa informada, onde o tipo de despesa (tbtipodespesa), seja do tipo informado como o segundo parâmetro da função (número). Neste caso, teste o atributo servicoproduto da tabela de tbtipodespesa . A função não poderá retornar NULL, desta forma caso não exista valor deverá retornar 0(zero).

**CREATE OR REPLACE FUNCTION** *get\_despesa\_veiculo*(  
 placa **IN** TBVEICULO.PLACA%**TYPE**,  
 tipo **IN INTEGER**)  
**RETURN NUMBER  
IS** valor **NUMBER**;  
**BEGIN  
  
 SELECT** *SUM*(dvei.VALORDESPESA) **INTO** valor **FROM** TBVEICULO vei  
 **INNER JOIN** TBDESPESASVEI dvei  
 **ON** vei.PKCODVEI = dvei.FKCODVEI  
 **INNER JOIN** TBTIPODESPESA tdes  
 **ON** dvei.FKTIPODESP = tdes.PKTIPODESP  
 **WHERE** vei.PLACA = placa **AND** tdes.SERVICOPRODUTO = tipo;  
  
 **IF** (valor **IS NULL** ) **THEN** valor := 0;  
 **END IF**;  
  
 **RETURN** valor;  
**END** *get\_despesa\_veiculo*;

1. Crie um procedimento que passada a placa, modelo e valor de compra e código da marca do veículo cadastre um novo veículo. Mostre dentro do procedimento o código que o veículo possuirá, o seu modelo, o nome da marca do veículo,e a quantidade de veículos vinculados a respectiva marca. O procedimento também deverá incrementar em mais um, a quantidade de veículos (quantveiculo) associada da tabela de marca

**CREATE OR REPLACE PROCEDURE** *cad\_veiculo*(  
 placa **IN** TBVEICULO.PLACA%**TYPE**,  
 modelo **IN** TBVEICULO.MODELO%**TYPE**,  
 valor\_compra **IN** TBVEICULO.VALORCOMPRA%**TYPE**,  
 cod\_marca **IN** TBVEICULO.FKCODMARCA%**TYPE**)  
**IS** cod\_veiculo **INTEGER**;  
 nome\_marca TBMARCA.NOMEMARCA%**TYPE**;  
 qtd\_vinculados **INTEGER**;  
**BEGIN  
 SELECT** *get\_prox\_cod\_cad*('TBVEICULO') **INTO** cod\_veiculo **FROM** dual;  
 **SELECT** NOMEMARCA, QUATVEICULOS **INTO** nome\_marca, qtd\_vinculados **FROM** TBMARCA **WHERE** PKCODMARCA = cod\_marca;  
  
 DBMS\_OUTPUT.PUT\_LINE('Código: ' || cod\_veiculo || ' - Modelo: ' || modelo || ' - Marca: ' || nome\_marca || ' - Veículos vinculados: ' || qtd\_vinculados);  
  
 **INSERT INTO** TBVEICULO (PKCODVEI, PLACA, MODELO, VALORCOMPRA, FKCODMARCA) **VALUES** (cod\_veiculo, placa, modelo, valor\_compra, cod\_marca);  
  
 **UPDATE** TBMARCA **SET** QUATVEICULOS = QUATVEICULOS + 1 **WHERE** PKCODMARCA = cod\_marca;  
**END** *cad\_veiculo*;

1. Crie um procedimento que registre uma despesa nova no veículo. Neste caso o procedimento conterá apenas os parâmetros: Placa do veículo; descrição da despesa; valor da despesa; data do registro da despesa; e código do tipo de despesa do veículo. O procedimento dever
   * Verifique se a placa passada como parâmetro está cadastrada e se o veículo não foi vendido.(o campo vendido na tabela de veículo armazena uma flag onde 0 representa que não foi vendido e 1 representa que foi vendido). Caso o veículo não exista insira o mesmo na tabela de veículo chamando o procedimento de inserção. Defina a marca deste novo veículo como sendo o código da maior marca cadastrada. Caso exista o veículo cadastrado busque o código do veículo cadastrado com a respectiva placa que não foi vendido. Se o veículo foi vendido, não execute os demais passos que seguem.
   * Em seguida armazena na tabela de “tbdespesasvei” um registro registrando a despesa do veículo
   * Altere os campos somatoriodespesas, quantdespesas e mediapordespesas da tabela “tbveículo” atualizando estas informações com os novos dados informados
   * Altere o campo somatoriodesp da tabela “tipodespesa” atualizando o mesmo com os dados inseridos

**CREATE OR REPLACE PROCEDURE** *reg\_despesa*(  
 placa **IN** TBVEICULO.PLACA%**TYPE**,  
 descricao **IN** TBDESPESASVEI.DESCRICAODESPESA%**TYPE**,  
 valor **IN** TBDESPESASVEI.VALORDESPESA%**TYPE**,  
 data\_reg **IN** TBDESPESASVEI.DATALANC%**TYPE**,  
 cod\_tipo **IN** TBTIPODESPESA.PKTIPODESP%**TYPE**)  
**IS** qtd\_veiculo **INTEGER**;  
 is\_vendido **INTEGER**;  
 cod\_vei **INTEGER**;  
 prox\_cod\_despesa **INTEGER**;  
 media\_desp **NUMBER**;  
**BEGIN  
 SELECT** *COUNT*(PKCODVEI) **INTO** qtd\_veiculo **FROM** TBVEICULO **WHERE** PLACA = placa;  
  
 **IF** (qtd\_veiculo = 0) **THEN** exec cad\_veiculo(placa, **NULL**, 0, (**SELECT MAX**(PKCODMARCA) **FROM** TBMARCA);  
 **END IF**;  
  
 **SELECT** VENDIDO, PKCODVEI **INTO** is\_vendido, cod\_vei **FROM** TBVEICULO **WHERE** PLACA = placa;  
  
 **IF** (is\_vendido = 0) **THEN  
 SELECT** get\_prox\_cod('TBVEICULO') **INTO** prox\_cod\_despesa **FROM** dual;  
  
 **INSERT INTO** TBDESPESASVEI **VALUES** (prox\_cod\_despesa, cod\_vei, cod\_tipo, descricao, data\_reg, valor);  
  
 **SELECT** MEDIAN(VALORDESPESA) **INTO** media\_desp **FROM** TDESPESASVEI **WHERE** FKCODVEI = cod\_vei;  
  
 **UPDATE** TBVEICULO **SET** SOMATORIODESPESAS = SOMATORIODESPESAS + valor, QUANTDESPESAS = QUANTDESPESAS + 1, MEDIAPORDESPESAS = media\_desp **WHERE** PKCODVEI=cod\_vei;  
  
 **UPDATE** TBTIPODESPESA **SET** SIMATORIODESP = SIMATORIODESP + valor **WHERE** PKTIPODESP = cod\_tipo;  
 **END IF**;  
**END** reg\_despesa;

Prof. Cadoná