

DATABASE APPLICATION & DATA SCIENCE

Capacitar o aluno a criar aplicações robustas, seguras e eficientes para manipulação de grandes volumes de dados, utilizando a linguagem ANSI SQL e PL/SQL em banco de dados Oracle. Você conhecerá elementos como: procedures, functions, packages, triggers, collections, PL/SQL tables, PL/SQL com Bulk processing, Autonomous transactions, Exception handling, SQL tuning.



Apresentação

Nome: Marcel Thomé Filho

Cargo: Professor

Titulação: Mestre

Área: BD – Modelagem, SQL, Programação....

Afins: IA, RN, DW, DM....

E-mail: profmarcel.filho@fiap.com.br

Coord. Curso: Etec Guarulhos



Agenda:

Revisão conteúdo aula passada

Estrutura de decisão

Exercícios

Migração de dados: tabela – bloco

Migração de dados: bloco – tabela

Recursos da linguagem

Estrutura
Em
blocos

Declare

/* declaração de variáveis de memória –
opcional

Begin

/* instruções de funcionamento –
processamento, ifs

Exception

/* tratamento de exceções
opcional

End

/* finalização do bloco

Recursos da linguagem

Estrutura De Decisão

If – then - elsif – then – else – end if

```
IF <condição> THEN
    <instruções>;
ELSIF <condição> THEN
    <instruções>;
ELSE
    <instruções>;
END IF;
```

Recursos da linguagem

Estrutura
De
Decisão

```
DECLARE
    V_N NUMBER(2) := 15;
BEGIN
    IF MOD(V_N,2) = 0 THEN
        DBMS_OUTPUT.PUT_LINE('O número ' || V_N || ' é PAR');
    ELSE
        DBMS_OUTPUT.PUT_LINE('O número ' || V_N || ' é ÍMPAR');
    END IF;
END;
```

Exercício para tomada de decisão

Criar um bloco PL/Sql para analisar a entrada de dados do sexo de um cliente, o bloco deverá receber o dado sobre o sexo: para masculino – M ou m, para feminino - F ou f, qualquer dado fora desta configuração deverá ser exibido 'Outros', para M ou m 'Masculino', para F ou f 'Feminino'.

Gabarito

```
declare
    v_sexo char(1) := '0';
begin
    if v_sexo = 'f' or v_sexo = 'F' then
        dbms_output.put_line('Feminino');
    elsif v_sexo = 'm' or v_sexo = 'M' then
        dbms_output.put_line('Masculino');
    else
        dbms_output.put_line('Outros');
    end if;
end;
```

Exercício para tomada de decisão

Criar um bloco PL-SQL para calcular o valor de cada parcela de uma compra de um carro, nas seguintes condições:

- Parcelas para aquisição em 6 pagamentos.
- Parcelas para aquisição em 12 pagamentos.
- Parcelas para aquisição em 18 pagamentos.

OBSERVAÇÃO:

- 1 - Deverá ser dada uma entrada de 20% do valor da compra.
 - 2 – Deverá ser aplicada uma taxa juros, no saldo restante, nas seguintes condições:
 - 3 – No final informar o valor das parcelas para as 3 formas de pagamento, com o Valor de aquisição de 10.000 e o mesmo com entrada de dados via teclado (em tempo de execução).
- A – Pagamento em 6 parcelas: 10%.
 - B – Pagamento em 12 parcelas: 15%.
 - C – Pagamento em 18 parcelas: 20%.

```
declare
v_carro number(10,2) := 50000 * 0.8;
v_presta v_carro%type;
v_numero number(2) := 10;
begin
    if v_numero = 6 then
        v_presta := (v_carro * 1.1) / 6;
        dbms_output.put_line('Valor da prestação em 6x: ' || v_presta);
    elsif v_numero = 12 then
        v_presta := (v_carro * 1.15) / 12;
        dbms_output.put_line('Valor da prestação em 12x: ' || v_presta);
    elsif v_numero = 18 then
        v_presta := (v_carro * 1.2) / 18;
        dbms_output.put_line('Valor da prestação em 18x: ' || v_presta);
    else
        dbms_output.put_line('Fora de contrato');
    end if;
end;
```

Instruções DML e DQL no bloco

Instrução select

```
SELECT NOME_DA_COLUNA INTO NOME_DA_VARIAVEL  
FROM NOME_DA_TABELA WHERE ...;
```

Instrução DQL, exemplo:

Tabela: aluno

RA	CHAR	9	Primary Key
NOME	VARCHAR2	50	

Instrução DQL, exemplo:

Tabela: aluno

```
CREATE TABLE ALUNO ( RA CHAR(9),  
NOME VARCHAR2(50),  
CONSTRAINT ALUNO_PK PRIMARY KEY(RA));
```

Instrução DQL, exemplo:

Tabela: aluno

111222333	Antonio Alves
222333444	Beatriz Bernardes
333444555	Cláudio Cardoso

Instrução DQL, exemplo:

Tabela: aluno

```
INSERT INTO ALUNO (RA,NOME) VALUES ('111222333','Antonio Alves');  
INSERT INTO ALUNO (RA,NOME) VALUES ('222333444','Beatriz Bernardes');  
INSERT INTO ALUNO (RA,NOME) VALUES ('333444555','Cláudio Cardoso');
```


Instrução DQL, exemplo:

Tabela: aluno

E, finalmente, vamos criar um bloco PL/SQL que deverá imprimir na tela o nome do aluno cujo RA é igual a 333444555:

Instrução DQL, exemplo:

```
DECLARE
    V_RA CHAR(9) := '333444555';
    V_NOME VARCHAR2(50);
BEGIN
    SELECT NOME INTO V_NOME FROM ALUNO WHERE RA = V_RA;
    DBMS_OUTPUT.PUT_LINE ('O nome do aluno é: ' || V_NOME);
END;
```

Instrução DML, exemplo:

```
DECLARE
    V_RA CHAR(9) := '444555666';
    V_NOME VARCHAR2(50) := 'Daniela Dorneles';
BEGIN
    INSERT INTO ALUNO (RA,NOME) VALUES (V_RA,V_NOME);
END;
```

Instrução DML, exemplo:

```
DECLARE
    V_RA CHAR(9) := '111222333';
    V_NOME VARCHAR2(50) := 'Antonio Rodrigues';
BEGIN
    UPDATE ALUNO SET NOME = V_NOME WHERE RA = V_RA;
END;
```

Instrução DML, exemplo:

```
DECLARE  
    V_RA CHAR(9) := '444555666';  
BEGIN  
DELETE FROM ALUNO WHERE RA = V_RA;  
END;
```

Tire suas Dúvidas

