

MASTERING RELATIONAL AND NON-RELATIONAL DATABASE

Durante este curso, você mergulhará na linguagem ANSI SQL e PL/SQL, desvendando sua versatilidade no contexto do banco de dados Oracle. Aprenderá sobre componentes cruciais, incluindo procedures, functions, packages, triggers, collections, PL/SQL tables, PL/SQL com Bulk processing, Autonomous transactions e Exception handling. Além disso, receberá orientações essenciais para aprimorar a otimização de consultas SQL. Navegue pelo mundo do MongoDB para aprofundar os conhecimentos em bancos de dados não relacionais. Do design sem esquema ao armazenamento flexível de documentos, você mergulhará no cerne do MongoDB e suas capacidades. Ganhe expertise em consultas usando a Linguagem de Consulta MongoDB (MQL), aprenda a realizar operações eficientes de CRUD e aproveite o poder dos pipelines de agregação para processamento avançado de dados.



Apresentação:

Nome: Diogo Alves

Cargo: Professor

Área: BD – Modelagem, SQL, NoSQL

Afins: Big Data, BI, DW, Programação...

E-mail: profdiogo.alves@fiap.com.br

Analista de Sistemas Sênior

Bem-vindos

Experimente

Nesta lição, você aprenderá:

- Revisão conteúdo aula passada
- Estrutura de decisão
- Exercícios
- Migração de dados: tabela > bloco
- Migração de dados: bloco > tabela



ESTRUTURA EM BLOCOS

Declare

/* declaração de variáveis de memória

– opcional */

Begin (Obrigatório)

/* instruções de funcionamento

– processamento, ifs e loops */

Exception

/* tratamento de exceções

Opcional */

End; (Obrigatório)

--finalização do bloco



RECURSOS DA LINGUAGEM

FIAP

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

ESTRUTURA
DE
DECISÃO

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





RECURSOS DA LINGUAGEM

FIAP

```
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;
```

ESTRUTURA
DE
DECISÃO



Bora fazer uns exercícios



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'.





BLOCO
DE
PL/SQL

```
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:

A – Pagamento em 6 parcelas: 10%.

B – Pagamento em 12 parcelas: 15%.

C – Pagamento em 18 parcelas: 20%.

3 – No final informar o valor das parcelas para as 3 formas de pagamento, com o Valor de aquisição e número de parcelas com entrada de dados via teclado (em tempo de execução).





GABARITO

FIAP

BLOCO DE PL/SQL

```
declare
v_carro number(10,2) := &vlCarro * 0.8;
v_presta v_carro%type;
v_numero number(2) := &nrParcela;
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;
```



Tire suas Dúvidas



Instruções DML e DQL no bloco

Instrução select

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

Instrução DDL, exemplo:

Tabela: aluno

PF1789.ALUNO	
RM	NUMBER (10)
NOME	VARCHAR2 (20 BYTE)
CPF	VARCHAR2 (11 BYTE)
SOBRENOME	VARCHAR2 (20 BYTE)

Instrução DML, exemplo:

Tabela: aluno

```
INSERT INTO ALUNO VALUES (1,'Antonio','9999999','Alves');  
INSERT INTO ALUNO VALUES (2,'Beatriz','8888888','Bernardes');  
INSERT INTO ALUNO VALUES (3,'Cláudio','7777777','Cardoso');
```

Instrução DQL, exemplo:

Tabela: aluno

E, finalmente, vamos criar um bloco PL/SQL que deverá imprimir na tela o nome, sobrenome e cpf do aluno cujo RM é igual a &RM:

Instrução DQL, exemplo:

DECLARE

```
V_RM NUMBER(10) := &RM;  
V_NOME VARCHAR2(20);  
V_SOBRE_NOME VARCHAR2(20);  
V_CPF VARCHAR2(11);
```

BEGIN

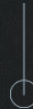
```
SELECT NOME, SOBRENOME, CPF  
  INTO V_NOME, V_SOBRE_NOME , V_CPF FROM ALUNO WHERE RM = V_RM ;  
DBMS_OUTPUT.PUT_LINE ('NOME: ' || V_NOME);  
DBMS_OUTPUT.PUT_LINE ('SOBRE NOME: ' || V_SOBRE_NOME );  
DBMS_OUTPUT.PUT_LINE ('CPF: ' || V_CPF );
```

END;

Instrução DML, exemplo:

Tabela: aluno

E, agora, vamos criar um bloco PL/SQL que deverá incluir na tabela aluno o RM e nome do aluno: 4, Daniela Dorneles.



Instrução DML, exemplo:

DECLARE

V_RM NUMBER(10) := 4;

V_NOME VARCHAR2(20) := 'Daniela';

V_SOBRE_NOME VARCHAR2(20) := 'Dorneles';

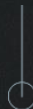
BEGIN

INSERT INTO ALUNO (RM,NOME, SOBRENOME)

VALUES (V_RM,V_NOME, V_SOBRE_NOME);

DBMS_OUTPUT.PUT_LINE ('Aluno incluído com sucesso');

END;



Instrução DML, exemplo:

Tabela: aluno

Vamos criar um bloco PL/SQL que deverá atualizar na tabela aluno o nome do aluno com RM: 1.

Instrução DML, exemplo:

DECLARE

V_RM NUMBER(10) := 1;

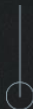
V_NOME VARCHAR2(20) := 'Alex';

BEGIN

UPDATE ALUNO SET NOME = V_NOME WHERE RM = V_RM;

DBMS_OUTPUT.PUT_LINE ('Aluno atualizado com sucesso');

END;



Instrução DML, exemplo:

Tabela: aluno

Vamos criar um bloco PL/SQL que deverá deletar na tabela aluno o aluno com RM: 4.

Instrução DML, exemplo:

DECLARE

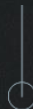
V_RM NUMBER(10) := 4;

BEGIN

DELETE FROM ALUNO WHERE RM = V_RM;

DBMS_OUTPUT.PUT_LINE ('Aluno deletado com sucesso');

END;





Tire suas Dúvidas

