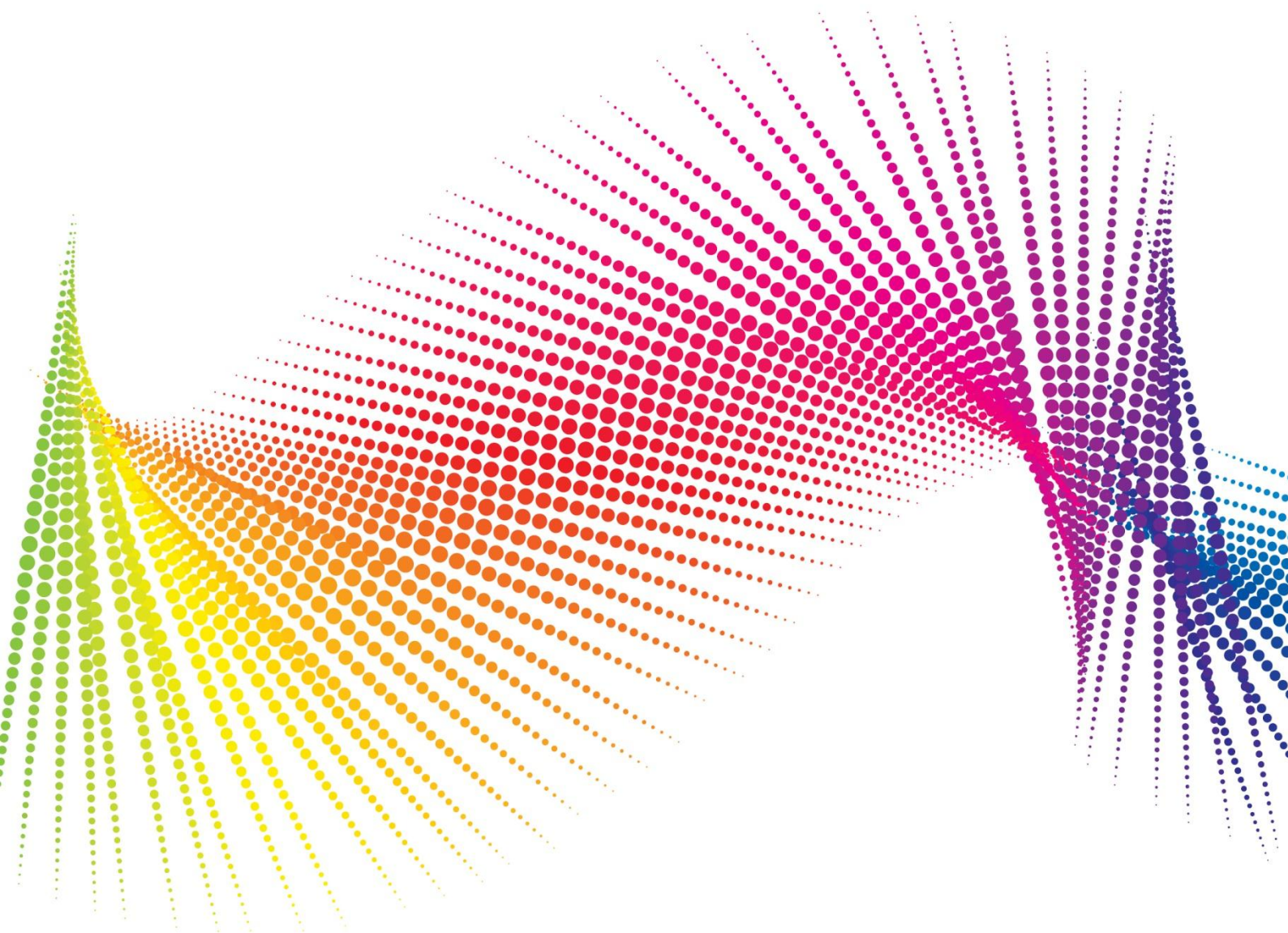


# Programação para Banco de Dados

Aula 01



Este material é parte integrante da disciplina, oferecida pela UNINOVE.

O acesso às atividades, conteúdos multimídia e interativo, encontros virtuais, fóruns de discussão e a comunicação com o professor devem ser feitos diretamente no ambiente virtual de aprendizagem UNINOVE.

Uso consciente do papel.

Cause boa impressão, imprima menos.

## **Aula 01: Introdução à linguagem PL/SQL**

**Objetivo:** Nesta primeira aula, apresentar os conceitos básicos e os principais recursos da linguagem PL/SQL.

### **Introdução**

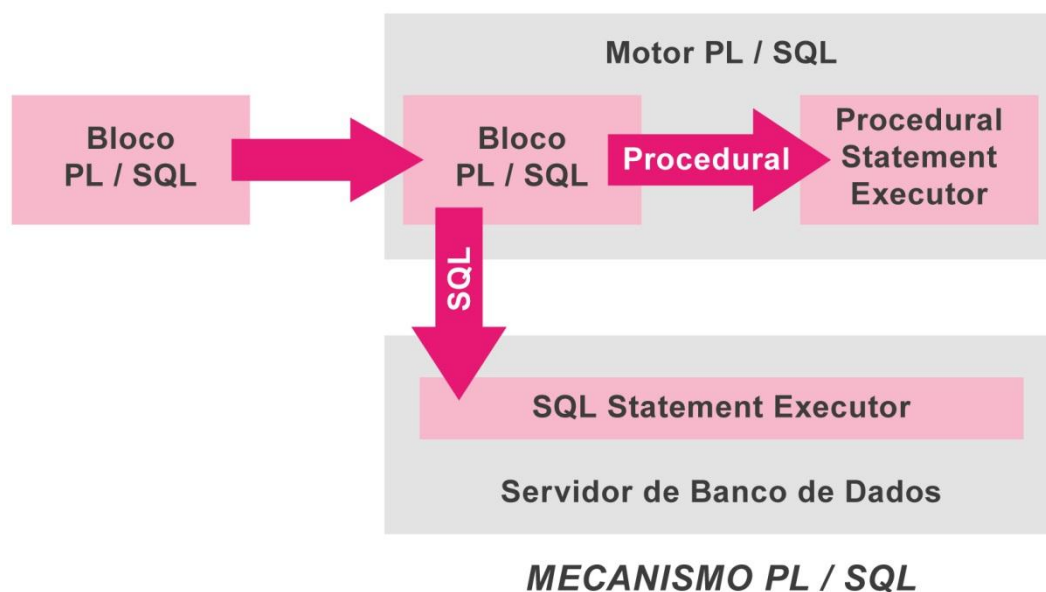
A PL/SQL é uma linguagem utilizada para acessar um banco de dados Oracle a partir de vários ambientes. Nesta primeira aula faremos uma breve apresentação e abordaremos os recursos mais importantes e as principais vantagens desta linguagem.

O acrônimo PL/SQL significa Procedural Language/ Structured Query Language. O nome justifica-se, pois a linguagem integra construções procedurais com o acesso ao banco de dados por meio da linguagem SQL.

É utilizada para acessar um banco de dados Oracle a partir de vários ambientes. Integrada com o servidor de banco de dados, permite que o código PL/SQL possa ser processado de forma rápida e eficiente.

A PL/SQL apresenta muita semelhança com a linguagem Ada que, por sua vez, é baseada no Pascal. A linguagem Ada, que recebeu este nome em homenagem à Ada Lovelace<sup>1</sup>, surgiu no final da década de 1970 e tornou-se padrão ANSI e ISO na década de 1980.

PL/SQL combina as vantagens da SQL com as construções procedurais de uma linguagem de terceira geração. A SQL é uma linguagem de quarta geração (4GL), descreve o que deve ser feito, mas não descreve como fazê-lo. Linguagens de terceira geração (3GL) são geralmente mais procedurais, implementam um algoritmo para resolver passo a passo determinado problema. Linguagens de quarta geração são mais simples e geralmente têm menos comandos. No entanto, em determinadas situações torna-se necessário recorrer às linguagens de terceira geração para solucionar problemas que não podem ser resolvidos apenas com a utilização de linguagens de quarta geração. A PL/SQL estende a SQL incluindo estruturas encontradas em outras linguagens: definição de tipos e variáveis, estruturas de decisão e de repetição, procedimentos e funções etc.



## Recursos da linguagem PL/SQL

Descreveremos a seguir brevemente os principais recursos oferecidos pela linguagem PL/SQL. Examinaremos com detalhes cada um deles durante as próximas aulas.

- **Estrutura de blocos**

Os programas desenvolvidos em PL/SQL são compostos por blocos. Cada um deles realiza uma unidade lógica de trabalho no programa e podem ser aninhados um dentro do outro.

Observe a seguir a estrutura básica de um bloco PL/SQL:

```
DECLARE
/* Nesta seção declaramos variáveis, tipos, cursores, etc.
BEGIN
/* Nesta seção inserimos instruções procedurais e SQL
EXCEPTION
/* Nesta seção tratamos as exceções
END;
```

- **Tratamento de erros**

Conforme observado no item anterior os blocos PL/SQL podem apresentar uma seção para tratamento de exceções, isto é, responder a erros encontrados durante a execução dos programas.

Há várias exceções pré-definidas pelo Oracle que podem ser tratadas nesta seção, mas o usuário pode criar outras exceções e tratá-las de forma apropriada em seus programas.

- **Tipos e variáveis**

Uma variável é uma posição de memória que pode ser atribuída ou lida pelo programa. Informações são transmitidas entre os programas PL/SQL e o banco de dados por meio de variáveis.

Cada variável apresenta um tipo específico que define qual informação a variável pode manter: um número inteiro, um número com ponto flutuante, uma data etc.

Além dos tipos definidos pela linguagem, a PL/SQL permite que o programador defina outros tipos personalizando dessa forma a estrutura dos dados que o seu programa manipula.

- **Estruturas de decisão**

A PL/SQL disponibiliza a construção IF ... THEN ... ELSE para implementar as estruturas de decisão ou condicionais. Utilizamos essas estruturas para executar partes diferentes de um programa dependendo do resultado de um teste, isto é, se uma determinada condição for verdadeira ou falsa. Os operadores relacionais (=, <, >, <=, >=, <>) são utilizados para realizar as comparações.

- **Estruturas de repetição**

Utilizamos as estruturas de repetição quando precisamos repetir um determinado trecho de um programa determinado número de vezes.

A linguagem PL/SQL permite que trabalhemos com laços contados, utilizados quando sabemos qual o número de vezes que o trecho do programa vai ser executado, ou laços condicionais, utilizados quando uma condição deve ser verificada para determinar até quando o trecho deverá ser executado.

- **Cursors**

Os cursores são utilizados para processar várias linhas recuperadas de um banco de dados por meio de uma instrução SELECT e manipular o resultado desta consulta linha a linha dentro de um bloco PL/SQL.

- **Procedimentos**

Procedimentos armazenados ou stored procedures são subprogramas que executam uma determinada ação. Não retornam valores e, portanto, não são utilizados para atribuir valores a variáveis ou como argumento em um comando SELECT. Os procedimentos opcionalmente podem receber argumentos de entrada que serão utilizados na execução do subprograma.

- **Funções**

Funções são subprogramas que executam uma determinada ação e retornam valores. Dessa maneira, diferentemente dos procedimentos armazenados, podem ser invocadas por meio de um comando SELECT ou utilizadas em cálculos. As funções também podem receber argumentos de entrada para que sejam utilizados na execução do subprograma.

- **Gatilhos**

Gatilhos ou triggers são blocos PL/SQL disparados automática e implicitamente sempre que ocorrer um evento associado a uma tabela (INSERT, UPDATE ou DELETE). São utilizados basicamente para manutenção de tabelas, implementação de níveis de segurança mais complexos, geração de valores de

colunas (exemplo: gerar o valor total do pedido a cada inclusão, alteração ou exclusão de um pedido).

- **Pacotes**

Pacotes ou packages são objetos do banco de dados semelhantes a bibliotecas, que armazenam procedimentos, funções, definições de cursores, variáveis, constantes e definições de exceções. Um pacote é composto de duas partes:

1. Especificação: área onde são feitas as declarações públicas. As variáveis, constantes, cursores, exceções e subprogramas estarão disponíveis para uso externo ao pacote.
2. Corpo: área onde são feitas as declarações privadas que estarão disponíveis apenas dentro do package e a definição de ações para os subprogramas públicos e privados.

- **Coleções**

As coleções PL/SQL são semelhantes aos arrays de outras linguagens de terceira geração. A PL/SQL oferece três tipos diferentes de coleções: varrays, tabelas index-by e tabelas aninhadas.

Antes de utilizar os recursos que acabamos de descrever, nas próximas três aulas, apresentaremos uma rápida revisão dos principais comandos da linguagem SQL. Primeiramente vamos relembrar alguns dos principais comandos que fazem parte do grupo denominado DDL – Data Definition Language: CREATE TABLE, ALTER TABLE e DROP TABLE. A seguir, veremos os comandos que fazem parte da DML – Data Manipulation Language: INSERT, UPDATE e DELETE. Concluindo esta etapa de revisão, vamos falar da DQL – Data Query Language, abordando o comando SELECT.

Chegamos ao final de nossa primeira aula. Agora acesse o AVA da UNINOVE e faça os exercícios propostos. Se ficar com dúvidas, não deixe de esclarecê-las com o seu professor.



**Saiba Mais**

(1) Ada Lovelace: filha de Lord Byron, considerada por muitos a primeira programadora, foi colaboradora de Charles Babbage, reverenciado como o inventor que projetou o primeiro computador de uso geral utilizando apenas partes mecânicas: a máquina analítica.

**REFERÊNCIAS**

FEUERSTEIN, Steven. *Linguagem PL/SQL Oracle*. Rio de Janeiro: Alta Books, 2006.

McLAUGHLIN, Michael. *Oracle Database 11g PL/SQL Programação*. Rio de Janeiro: Alta Books, 2009.

MORELLI, Eduardo Terra. *Oracle 9i fundamental SQL, PL/SQL e Administração*. São Paulo: Érica, 2005.

RAMALHO, José Antonio. *Oracle 10g*. São Paulo: Thomson, 2005.

URMAN, Scott. *Oracle 9i programação PL/SQL*. Rio de Janeiro: Elsevier, 2002.