

FIAP

FIAP GRADUAÇÃO

Tecnologia em Análise e Desenvolvimento de Sistemas

Prof^o Ms. Alexandre Barcelos
profalexandre.barcelos@fiap.com.br

2019

Database Application Development

Prof^o Ms. Alexandre Barcelos
profalexandre.barcelos@fiap.com.br

2019



Introdução ao PL/SQL

- **Explicar a definição de PL/SQL**
- **Explicar os benefícios do PL/SQL**
- **Identificar os diferentes tipos de blocos PL/SQL**

Objetivo da Lição

Esta lição apresenta as regras e estruturas básicas para criação e execução dos blocos de códigos PL/SQL. Ela também mostra como declarar variáveis e atribuir tipos de dados a elas.

PL/SQL:

- É uma extensão para a linguagem SQL com facilidades projetadas de linguagens de programação.
- Declarações de Manipulação de Dados e Consultas SQL são incluídas dentro de unidades de código procedural.



PL/SQL:

- **Fornece uma estrutura de bloco para unidades de código executáveis. A manutenção do código é facilitada com uma estrutura bem definida.**
- **Fornece construções procedurais tais como:**
 - **Variáveis, constantes e tipos**
 - **Estruturas de controle, como declarações condicionais e loops**
 - **Unidades de programa reutilizáveis que são escritas uma vez e executadas várias vezes**

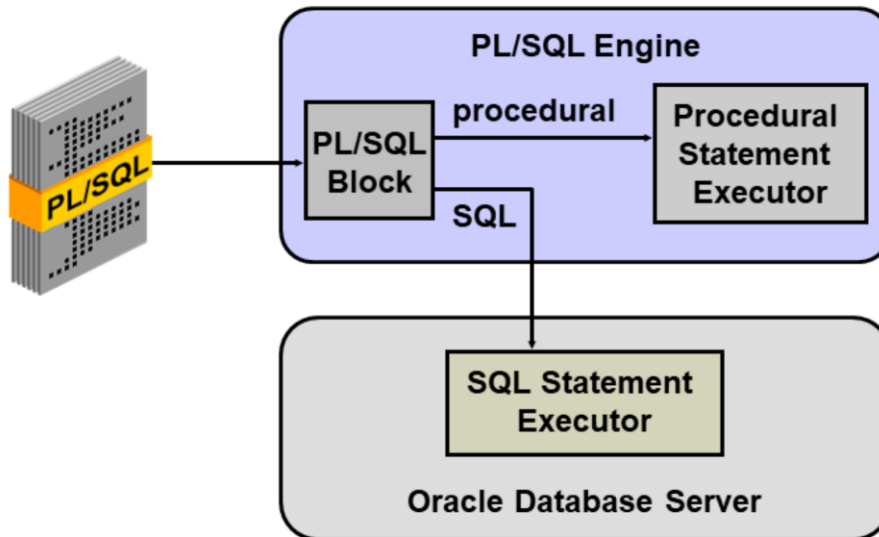
1-7

Sobre PL/SQL

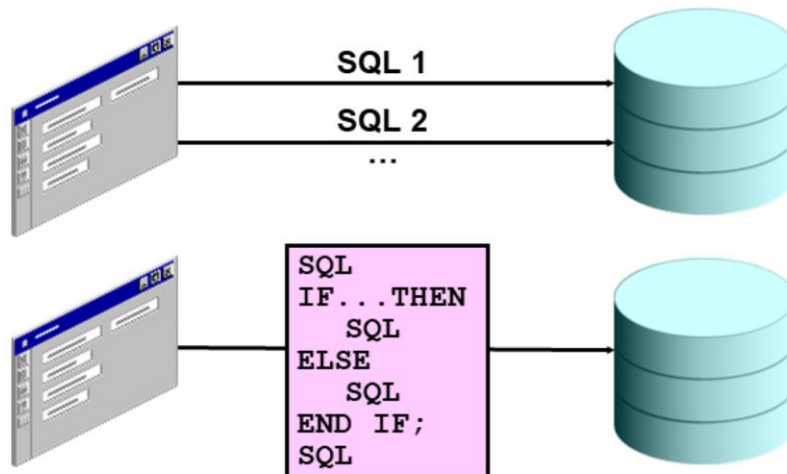
A linguagem PL/SQL (Procedural Language/SQL) é uma extensão de linguagem procedural da Oracle Corporation para SQL, a linguagem de acesso a dados padrão para bancos de dados relacionais. A linguagem PL/SQL oferece recursos de engenharia de software modernos, como, por exemplo, a encapsulação de dados, o tratamento de exceções, a ocultação de informações e a orientação a objeto, etc., trazendo os recursos de programação mais modernos para o Oracle Server e o Toolset.

A linguagem PL/SQL incorpora muitos recursos avançados criados em linguagens de programação projetadas durante as décadas de 70 e 80. Além de aceitar a manipulação de dados, ele também permite que as instruções de consulta da linguagem SQL sejam incluídas em unidades procedurais de código e estruturadas em blocos, tornando a linguagem SQL uma linguagem avançada de processamento de transações. Com a linguagem PL/SQL, você pode usar as instruções SQL para refinar os dados do Oracle e as instruções de controle PL/SQL para processar os dados.

Ambiente PL/SQL



- Integração de construções procedurais com SQL
- Performance melhorada



1-9

Integração

A linguagem PL/SQL desempenha um papel central tanto para o Oracle Server (através de procedimentos armazenados, funções armazenadas, gatilhos de banco de dados e pacotes) quanto para as ferramentas de desenvolvimento Oracle (através de gatilhos de componente do Oracle Developer).

As aplicações do Oracle Developer fazem uso das bibliotecas compartilhadas que armazenam código (procedimentos e funções) e que podem ser acessadas local ou remotamente. O Oracle Developer consiste no Oracle Forms, Oracle Reports e Oracle Graphics.

Os tipos de dados SQL também podem ser usados no código PL/SQL. Combinados com o acesso direto que a linguagem SQL fornece, esses tipos de dados compartilhados integram a linguagem PL/SQL com o dicionário de dados do Oracle Server. A linguagem PL/SQL une o acesso conveniente à tecnologia de banco de dados com a necessidade de capacidade de programação procedural.

PL/SQL nas Ferramentas Oracle

Várias ferramentas Oracle, inclusive o Oracle Developer, têm o seu próprio

mecanismo PL/SQL, o qual é independente do mecanismo presente no Oracle Server.

O mecanismo filtra as instruções SQL e as envia individualmente ao executor da instrução SQL no Oracle Server. Ele processa as instruções procedurais restantes no executor da instrução procedural, que está no mecanismo PL/SQL.

O executor da instrução procedural processa os dados que são locais para a aplicação (que já estão no ambiente do cliente, em vez de estarem no banco de dados). Isso reduz o trabalho enviado ao Oracle Server e o número de cursores de memória necessários.

- **Desenvolvimento de programas modulares**
- **Integração com ferramentas Oracle**
- **Portabilidade**
- **Manipulação de exceções**

Desempenho Melhorado

A linguagem PL/SQL pode melhorar o desempenho de uma aplicação. Os benefícios diferem dependendo do ambiente de execução.

- A linguagem PL/SQL pode ser usado para agrupar as instruções SQL em um único bloco e enviar esse bloco inteiro para o servidor em uma única chamada, reduzindo assim o tráfego da rede. Sem o código PL/SQL, as instruções SQL seriam enviadas ao Oracle Server uma de cada vez. Cada instrução SQL resulta em uma outra chamada do Oracle Server e uma maior sobrecarga de desempenho. Em um ambiente de rede, a sobrecarga pode se tornar significativa.
- A linguagem PL/SQL também pode cooperar com as ferramentas de desenvolvimento de aplicação do Oracle Server como, por exemplo, Oracle Developer Forms e Reports. Ao

adicionar recursos de processamento procedural a essas ferramentas, a linguagem PL/SQL aumenta o desempenho.

Observação: Os procedimentos e as funções declaradas como parte de uma aplicação do Developer são distintos daqueles armazenados no banco de dados, embora as suas estruturas gerais sejam as mesmas. Os subprogramas armazenados são objetos de banco de dados e estão armazenados no dicionário de dados. Eles podem ser acessados por qualquer número de aplicações, incluindo as aplicações do Developer.

DECLARE (Opcional)

Variáveis, cursores, exceções
definidas pelo usuário

BEGIN (Obrigatório)

- Declarações SQL
- Declarações PL/SQL

EXCEPTION (Opcional)

Ações a serem executadas
quando ocorrem erros

END; (Obrigatório)



1-11

Estrutura de Bloco PL/SQL

A linguagem PL/SQL é uma linguagem estruturada em blocos, o que significa que os programas podem ser divididos em blocos lógicos. Um bloco PL/SQL consiste em até três seções: declarativa (opcional), executável (necessária) e tratamento de exceção (opcional). Apenas as palavras-chave BEGIN e END são necessárias. Você poderá declarar variáveis localmente para o bloco que as usa.

As condições de erro (conhecidas como exceções) podem ser tratadas especificamente no bloco aos quais elas se aplicam. Você poderá armazenar e alterar os valores em um bloco PL/SQL declarando e referenciando variáveis e outros identificadores.

A tabela a seguir descreve as três seções de bloco:

Executando Instruções e Blocos PL/SQL

Seção	Descrição	Inclusão
Declarativa	Contém todas as variáveis, constantes, cursores e exceções definidas pelo usuário que são referenciadas nas seções executável e declarativa	Opcional
Executável	Contém instruções SQL para manipular dados no banco de dados e instruções PL/SQL para manipular dados no bloco	Obrigatória
Tratamento de exceção	Especifica as ações a desempenhar quando erros e condições anormais surgem na seção executável	Opcional

As palavras-chave de seção DECLARE, BEGIN e EXCEPTION não são seguidas por ponto-e-vírgulas.

Entretanto, END e todas as outras instruções PL/SQL necessitam de um ponto-e-vírgula para terminar a instrução. Você poderá criar uma string de instruções na mesma linha, mas esse método não é recomendado pela clareza ou edição.

Anonymous

```
[DECLARE]

BEGIN
    --statements

[EXCEPTION]

END;
```

Procedure

```
PROCEDURE name
IS

BEGIN
    --statements

[EXCEPTION]

END;
```

Function

```
FUNCTION name
RETURN datatype
IS

BEGIN
    --statements
    RETURN value;

[EXCEPTION]

END;
```

Tipos de Bloco

Toda unidade PL/SQL compreende um ou mais blocos. Esses blocos podem ser inteiramente separados ou aninhados um dentro do outro. As unidades básicas (procedimentos e funções, também conhecidas como *subprogramas* e blocos anônimos) que compõem um programa PL/SQL são blocos lógicos, os quais podem conter qualquer número de sub-blocos aninhados. Por isso, um bloco pode representar uma pequena parte de um outro bloco, o qual, por sua vez pode ser parte da unidade de código inteira.

Dos dois tipos de construções PL/SQL disponíveis, blocos anônimos e subprogramas, apenas os blocos anônimos são abordados neste curso.

Blocos Anônimos

Os blocos anônimos são blocos sem nome. Eles são declarados em um ponto do aplicativo onde eles devem ser executados e são passados para o mecanismo PL/SQL para serem executados em tempo de execução. Você poderá incorporar um bloco anônimo em um programa pré-compilador. Os gatilhos nos componentes do Oracle Developer consistem nesses blocos.

Subprogramas

Os subprogramas são blocos PL/SQL nomeados que podem assumir parâmetros e podem ser chamados. Você pode declará-los como procedimentos ou como funções. Geralmente, você deve usar um procedimento para desempenhar uma ação e uma função para calcular um valor.

Você poderá armazenar subprogramas no servidor ou no nível de aplicação. Ao usar os componentes do Oracle Developer (Forms, Relatórios e Gráficos), você poderá declarar os procedimentos e funções como parte da aplicação (um form ou relatório) e chamá-los a partir de outros procedimentos, funções e gatilhos (veja a próxima página) na mesma aplicação sempre que necessário.

Observação: Uma função é similar a um procedimento, exceto que uma função *deve* retornar um valor.

Os procedimentos e funções são abordados no próximo curso sobre PL/SQL.

Construções de Programa

A tabela a seguir descreve em linhas gerais uma variedade de construções de programa PL/SQL que usam o bloco PL/SQL básico. Eles estão disponíveis de acordo com o ambiente no qual eles são executados.

- Ativar a com o comando
`SET SERVEROUTPUT ON`
- Use o pacote Oracle predefinido e seu procedimento:
- `DBMS_OUTPUT.PUT_LINE`

```
SET SERVEROUTPUT ON
...
DBMS_OUTPUT.PUT_LINE(' The First Name of the
Employee is ' || f_name);
...
```

1-15

Em um bloco PL/SQL, referencie o procedimento `DBMS_OUTPUT.PUT_LINE` e, entre parênteses, as informações que você deseja imprimir na tela. O pacote deve primeiro ser ativado na sessão. Para fazer isso, execute comando `SET SERVEROUTPUT ON`

Exemplo

Esse script calcula o salário mensal e imprime-as na tela, usando o procedimento `DBMS_OUTPUT.PUT_LINE`.

```
SET SERVEROUTPUT ON

ACCEPT p_annual_sal PROMPT 'Please enter the annual salary: '

DECLARE

v_sal NUMBER(9,2) := &p_annual_sal;

BEGIN

v_sal := v_sal/12;
```

```
DBMS_OUTPUT.PUT_LINE ('The monthly salary is ' ||  
TO_CHAR(v_sal));  
END;  
/
```



Esta prática abrange os seguintes tópicos:

- Identificar quais blocos PL/SQL são executados com êxito
- Criar e executar um bloco simples PL/SQL

Exercício

1. Qual dos seguintes blocos PL / SQL é executado corretamente?
 - a.

```
BEGIN  
END;
```
 - b.

```
DECLARE  
amount INTEGER(10);  
END;
```
 - c.

```
DECLARE  
BEGIN  
END;
```
 - d.

```
DECLARE  
amount INTEGER(10);  
BEGIN  
DBMS_OUTPUT.PUT_LINE(amount);  
END;
```
2. Crie e execute um bloco anônimo simples que exiba "Hello World". Execute e salve este script como lab_01_02_soln.sql.

