CONTENT INCLUDE:

- 1. INTRODUCTION
- 2. FEATURES OF PL/SQL
- 3. PL/SQL DATA TYPE
- 4. PL/SQL BLOCK STRUCTURE

INTRODUCTION TO PL/SQL:

- PL/SQL stands for "Procedural Language extensions to SQL”.
- The purpose of **PL/SQL** is to combine database language and procedural programming language.
- PL/SQL also enables you to define triggers, which are subprograms that the database executes automatically in response to specified events.
- SQL + PROCEDURAL FEATURES OF PROGRAMMING LANGUAGE=PL/SQL.

FEATURES OF PL/SQL:

- PL/SQL is tightly integrated with SQL.
- It supports object-oriented programming.
- It offers numerous data types.
- It supports developing web applications and server pages.

PL/SQL BLOCK STRUCTURE:

PL/SQL Block consists of three sections:

- The Declaration section (optional).
- The Execution section (mandatory).
- The Exception Handling (or Error) section (optional).

```
[DECLARE]
   Declaration statements;
BEGIN
   Execution statements;
 [EXCEPTION]
      Exception handling statements;
END;
/
```

SIMPLE EXAMPLE:

```
BEGIN
   NULL;
END
```

If you execute the above anonymous block in *SQL*Plus* you will see that it issues a message Saying:

“PL/SQL procedure successfully completed.”

Because the NULL statement does nothing.

EXAMPLE TO PRINT HELLO :

The following example displays a message Hello PL/SQL on a screen using SQL*Plus:

```
SET SERVEROUTPUT ON SIZE 1000000
BEGIN
  DBMS_OUTPUT.PUT_LINE('Hello PL/SQL');
END;
/
```

PL/SQL DATA TYPE:

- **Every variable has a data type(also called a type)** that determines its storage format, constraints, valid range of values, and operations that can be performed on it.

- PL/SQL provides many predefined data types and its subtype. A **subtype** is a subset of another data type, which is called its **base type.**

THESE ARE AS FOLLOWS:

1. CHARACTER data type
2. Number data type
3. BOOLEAN data type
4. DATE data type

1. CHARACTER DATA TYPE:

- This data type basically stores alphanumeric characters in string format.
- The literal values should always be enclosed between single quotes while assigning them to CHARACTER data type.
- This character data type is further classified as follows:

  1.1CHAR Data type (fixed string size)

  1.2VARCHAR2 Data type (variable string size)

  1.3VARCHAR Data type

1.1 CHAR DATA TYPE:

- This data type stores the string value, and the size of the string is fixed at the time of declaring the variable.

- The size restriction for this data type is 1-2000 bytes.

- CHAR data type is more appropriate to use where ever fixed size of data will be handled. .

- **Syntax for declaration:**

  employee CHAR;

  employee CHAR(10):=’Demo12’;

1.2 VARCHAR2 DATA TYPE:

- This data type stores the string, but the length of the string is not fixed.
- The size restriction for this data type is 1-4000 bytes for table column size and 1-32767 bytes for variables.
- The size is defined for each variable at the time of variable declaration.

Syntax for declaration:

employee VARCHAR2(10):=’Demo123’;

1.3 VARCHAR DATA TYPE:

- This is the synonymous with the VARCHAR2 data type.
- It is always a good practice to use VARCHAR2 instead of VARCHAR to avoid behavioral changes.

Syntax for declaration:

```
employee VARCHAR(10)='Demo12';
```

2. NUMBER DATA TYPE:

- This data type is used to work with fields which will contain only number data.

Syntax for declaration:

```
        NUMBER (8,2);
        NUMBER(8);
        NUMBER;
```

3. BOOLEAN DATA TYPE:

- It represents either TRUE or FALSE and mainly used in conditional statements. Values need not enclose within quotes while assigning for this data type.

Syntax for declaration:

```
    Var1 BOOLEAN;
```

4. DATE DATA TYPE:

- This data type stores the values in date format, as date, month, and year. Values need to enclose within quotes while assigning for this data type.
- The standard oracle time format for input and output is

  'DD-MON-YYYY’.

Syntax for declaration:

```
Active_date  DATE:=’03-FEB-1999’;
Today_date DATE:=SYSDATE;
```

THANK YOU