**Features of PL/SQL**

PL/SQL has the following features:

PL/SQL is tightly integrated with SQL.

It offers extensive error checking.

It offers numerous data types.

It offers a variety of programming structures.

It supports structured programming through functions and procedures.

It supports object-oriented programming.

It supports the development of web applications and server pages.

**Advantages of PL/SQL**

PL/SQL has the following advantages:

SQL is the standard database language and PL/SQL is strongly integrated with SQL.

PL/SQL supports both static and dynamic SQL. Static SQL supports DML operations

and transaction control from PL/SQL block. In Dynamic SQL, SQL allows embedding

DDL statements in PL/SQL blocks.

PL/SQL allows sending an entire block of statements to the database at one time.

This reduces network traffic and provides high performance for the applications.

PL/SQL gives high productivity to programmers as it can query, transform, and

update data in a database.

PL/SQL saves time on design and debugging by strong features, such as exception

handling, encapsulation, data hiding, and object-oriented data types.

Applications written in PL/SQL are fully portable.

PL/SQL provides high security level.

PL/SQL provides access to predefined SQL packages.

PL/SQL provides support for Object-Oriented Programming.

PL/SQL provides support for developing Web Applications and Server Pages.

**What is PL/SQL?**

PL/SQL is not a standalone programming language. **PL/SQL** is a **block-structured** language; this means that the PL/SQL programs are divided and written in logical blocks of code. Each block consists of three sub-parts:

1. Declarations
2. Executable Commands
3. Exception Handling

*(To run PL/SQL programs, you should have the Oracle BMS Server installed in your machine. This will take care of the execution of the SQL commands. The most recent version of Oracle RDBMS is 11g.)*

**N.B.** Every PL/SQL statement ends with a semicolon **(;)**. PL/SQL blocks can be nested within

other PL/SQL blocks using **BEGIN** and **END**.

**What is PL/SQL Identifier?**

PL/SQL identifiers are constants, variables, exceptions, procedures, cursors, and reserved

words. The identifiers consist of a letter optionally followed by more letters, numerals,

dollar signs, underscores, and number signs and should not exceed 30 characters.

By default, **identifiers are not case-sensitive**.

**N.B.** The PL/SQL supports single-line and multi-line comments. The PL/SQL single-line comments start with the delimiter **--** (double hyphen) and multi-line comments are enclosed by **/\*** and **\*/**.

**What are the PL/SQL Program Units?**

**PL/SQL Program Units**

A PL/SQL unit is any one of the following:

PL/SQL block

Function

Package

Package body

Procedure

Trigger

Type

Type body

**What is subtype?**

A subtype is a subset of another data type, which is called its base type. A subtype has

the same valid operations as its base type, but only a subset of its valid values. PL/SQL predefines several subtypes in package **STANDARD**. For example, PL/SQL predefines the subtypes **CHARACTER** and **INTEGER.**

**What is Null in PL/SQL?**

PL/SQL NULL values represent **missing** or **unknown data** and they are not an integer, a

character, or any other specific data type. Note that **NULL** is not the same as an empty

data string or the null character value **'\0'**. A null can be assigned but it cannot be equated

with anything, including itself.

**What is Variable?**

A variable is nothing but a name given to a storage area that our programs can manipulate. Each variable in PL/SQL has a specific data type, which determines the size and the layout of the variable's memory; the range of values that can be stored within that memory and the set of operations that can be applied to the variable.

**N.B. (**The name of a PL/SQL variable consists of a letter optionally followed by more letters, numerals, dollar signs, underscores, and number signs and should not exceed 30 characters. By default, variable names are not case-sensitive. You cannot use a reserved PL/SQL keyword is a variable name.**)**

**How to declare variable in PL/SQL?**

PL/SQL variables must be declared in the declaration section or in a package as a global variable. When you declare a variable, PL/SQL allocates memory for the variable's value and the storage location is identified by the variable name. The syntax for declaring a variable is:

**variable\_name [CONSTANT] datatype [NOT NULL] [:= | DEFAULT initial\_value]**

**What is Constants?**

A constant holds a value that once declared, does not change in the program. A constant declaration specifies its name, data type, and value, and allocates storage for it. The declaration can also impose the **NOT NULL constraint**.

**How to declare a Constant?**

A constant is declared using the **CONSTANT** keyword. It requires an initial value and does

not allow that value to be changed.

**What is Literal?**

A literal is an explicit numeric, character, string, or Boolean value not represented by an identifier. For example, TRUE, 786, NULL, 'tutorialspoint' are all literals of type Boolean, number, or string. PL/SQL, literals are case-sensitive. PL/SQL supports the following kinds of literals:

Numeric Literals

Character Literals

String Literals

BOOLEAN Literals

Date and Time Literals