Assignment No:-4

Title:

Unnamed PL/SQL code block: Use of Control structure and Exception handling is mandatory. Write a PL/SQL block of code for the following requirements:- Schema:

1.Borrower(Rollin, Name, DateofIssue, NameofBook, Status)

2.Fine(Roll_no,Date,Amt)

- Accept roll_no & name of book from user.
- Check the number of days (from date of issue), if days are between 15 to 30 then fineamount will be Rs 5 per day.
- If no. of days>30, per day fine will be Rs 50 per day & for days less than 30, Rs.5 perday.
- After submitting the book, status will change from I to R.
- If condition of fine is true, then details will be stored into fine table.

Objective:

- To study PL/SQL Block Writing

Theory:

What is PL/SQL block?

PL/SQL is the procedural approach to SQL in which a **direct instruction** is given to the PL/SQL engine about how to perform actions like storing/fetching/processing data. These instructions are **grouped** together called **Blocks**.

Blocks contain both PL/SQL as well as SQL instruction. All these instruction will be executed as a whole rather than executing a single instruction at a time.

Block Structure

PL/SQL blocks have a pre-defined structure in which the code is to be grouped. Beloware different sections of PL/SQL blocks

1. Declaration section
2. Execution section
3. Exception-Handling section

Declaration Section:

This is the first section of the PL/SQL blocks. This section is an optional part. This is the section in which the declaration of variables, cursors, exceptions, subprograms, instructions and collections that are needed in the block will be declared.

Below are few more characteristics of this part.

- This particular section is optional and can be skipped if no declarations are needed.This should be the first section in a PL/SQL block, if present.
- This section starts with the keyword 'DECLARE' for triggers and anonymous block. For other subprograms this keyword will not be present, instead the part after the subprogram name definition marks the declaration section.
- This section should be always followed by execution section.

Execution Section:

Execution part is the main and mandatory part which actually executes the code that is written inside it. Since the PL/SQL expects the executable statements from this block this cannot be an empty block, i.e., it should have at least one valid executable code line in it.

Below are few more characteristics of this part.

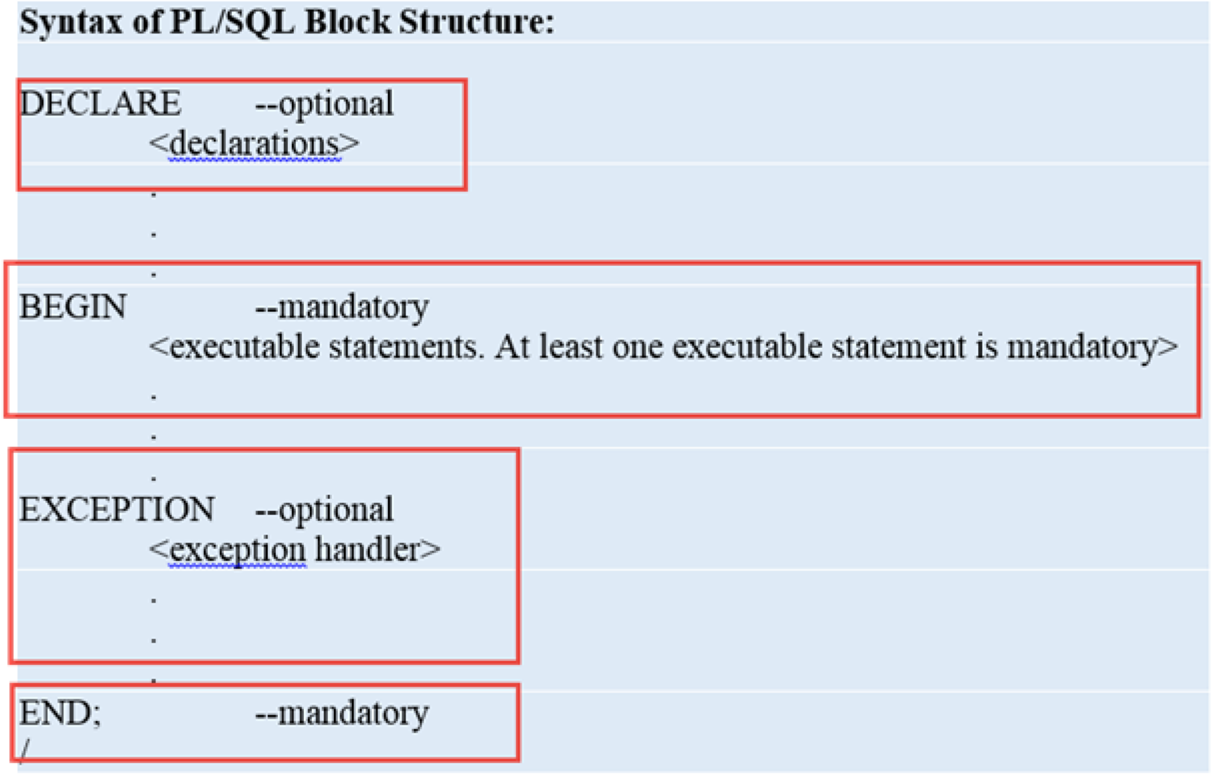
- This can contain both PL/SQL code and SQL code.
- This can contain one or many blocks inside it as nested blocks.This section starts with the keyword 'BEGIN'.
- This section should be followed either by 'END' or Exception-Handling section(ifpresent)

Exception-Handling Section:

The exception is unavoidable in the program which occurs at run-time and to handle this Oracle has provided an Exception-handling section in blocks. This section can also contain PL/SQL statements. This is an optional section of the PL/SQL blocks.

- This is the section where the exception raised in the execution block ishandled.
- This section is the last part of the PL/SQL block.
- Control from this section can never return to the execution block
- This section starts with the keyword 'EXCEPTION'.
- This section should be always followed by the keyword 'END'.

The Keyword 'END' marks the end of PL/SQL block. Below is the syntax of thePL/SQL block structure.



Syntax of PL/SQL Block Structure:

```
DECLARE         --optional
          <declarations>
          .
          .
          .
BEGIN           --mandatory
          <executable statements. At least one executable statement is mandatory>
          .
          .
          .
EXCEPTION       --optional
          <exception handler>
          .
          .
          .
END;            --mandatory
/
```

Note: A block should be always followed by '/' which sends the information to the compilerabout the end of the block. -

Types of PL/SQL block:

PL/SQL blocks are of mainly two types.

1. Anonymous blocks
2. Named Blocks

Anonymous blocks:

Anonymous blocks are PL/SQL blocks which do not have any names assigned to them. They need to be created and used in the same session because they will not be stored in the server as adatabase objects.

Since they need not to store in the database, they need no compilation steps. They are written andexecuted directly, and compilation and execution happen in a single process.

Below are few more characteristics of Anonymous blocks.

- These blocks don't have any reference name specified for them. These blocks start with the keyword 'DECLARE' or 'BEGIN'.
- Since these blocks are not having any reference name, these cannot be stored for later purpose. They shall be created and executed in the same session.
- They can call the other named blocks, but call to anonymous block is not possible asit is not having any reference.
- It can have nested block in it which can be named or anonymous. It can also be nested toany blocks.
- These blocks can have all three sections of the block, in which execution section is mandatory; the other two sections are optional.

Named blocks:

Named blocks are having a specific and unique name for them. They are stored as the database objects in the server. Since they are available as database objects, they can be referred to or usedas long as it is present in the server. The compilation process for named blocks happens separately while creating them as a database objects.

- These blocks can be called from other blocks.
- The block structure is same as an anonymous block, except it will never start with the keyword ‘DECLARE’. Instead, it will start with the keyword ‘CREATE’ which instruct thecompiler to create it as a database object.
- These blocks can be nested within other blocks. It can also contain nested blocks.Named blocks are basically of two types:

1. Procedure
2. Function

Conclusion:

After this section, we would be aware of PL/SQL blocks and its types, different sections of blocksand their usages.

Questions:

1. What is PL/SQL? Explain.
2. What is the difference between "SQL" and "PL/SQL"?