Introduction To PL-SQL ====> `set serveroutput on`
==========

* The PL/SQL stands for Procedural Language extensions to Structured Query Language.

* Basically, SQL is used to perform basic operations like:
- Creating a database
- Storing data in the database
- Updating data in the database
- Retrieving the stored data of database etc

* On the other hand PL/SQL is a fully Structured Procedural language which enables the developer to combine the powers of SQL with its procedural statements , to create super powerful SQL queries.

* This enhances the security, portability, & robustness of the database as well as gives more control to the programmers by the use of loops, conditions & object-oriented concepts.

Drawbacks Of SQL ===>
==========

* SQL does not provide the programming techniques of condition checking, looping and branching which is very important.

* SQL has no facility of error handling.

* SQL statements are passed to the Oracle Engine one at a time.

* While executing SQL statement, each time a call is made to the engine's resources.

* This increases traffic in the network and decreases the speed of data processing especially in a multi-user environment.

Advantages Of PL-SQL ===>
==========

* `Reduces network traffic` :- This is one of the greatest advantages of PL/SQL. Since PL/SQL allows us to club an entire block of SQL statements for execution into oracle engine , all at once , so it helps us in reducing the network traffic.

* `Procedural language support` :- PL/SQL provides the conditional checking, looping or branching operations just like any programming language.

* `Error handling` :- It provides facility to deal with errors, as required and displays user-friendly messages when error occurs.

* `Declare variable` :- PL-SQL allows declaration and use of variables in blocks of code. These variables are used to store intermediate results of a query for later processing either in SQL or PL/SQL.

* `Portable application` :- Applications written in PL/SQL are portable to any computer hardware and operating system, where Oracle is operational.

PL-SQL Architecture ====>
==========

* The PL/SQL architecture mainly consists of following three components:
- PL/SQL block
- PL/SQL Engine
- Database Server

PL-SQL Block ==>
==========

* This is the component which has the actual PL/SQL code.

* It consists of different sections to divide the code logically (declarative section for declaring purpose, execution section for processing statements, exception handling section for handling errors).

* It also contains the SQL instruction that used to interact with the database server.

* All the PL/SQL units are treated as PL/SQL blocks, and this is the starting stage of the architecture which serves as the primary input.

PL-SQL Engine ==>
==========

* PL/SQL engine is the component where the actual processing of the codes takes place.

* PL/SQL engine separates PL/SQL units and SQL part in the input.

* The separated PL/SQL units will be handled by the PL/SQL engine itself.

* The SQL part will be sent to database server where the actual interaction with database takes place.

Database Server ==>
==========

* This is the most important component of Pl/SQL unit which stores the data.

* The PL/SQL engine uses the SQL from PL/SQL units to interact with the database server.

* It consists of SQL executor which parses the input SQL statements and execute the same.

The PL-SQL Block ===>
==========

* PL/SQL program units organize the code into blocks.

* A block without a name is known as an anonymous block.

* The anonymous block is the simplest unit in PL/SQL.

* It is called anonymous block because it is not saved in the Oracle database.

DECLARE ==>
==========

* Variables and constants are declared, initialized within this section.

BEGIN ==>
==========

* This block is a procedural statement block whichwill implement the actual programming logic. This section contains conditional statements (if...else), looping statements (for, while) etc.

EXCEPTION ==>
==========

* A PL/SQL block has an exception-handling section that starts with the keyword EXCEPTION. The exception-handling section is where we catch and handle exceptions raised by the code in the execution section.

Example ==>

* The following example shows a simple PL/SQL anonymous block with one executable section:

```
- begin
  2  dbms_output.put_line('Welcome to PL-SQL');
  3  end;
  4  /
Welcome to PL-SQL
```

Code Explained ==>

* The SET SERVEROUTPUT ON command is not a PL-SQL command but it is used to instruct SQL*Plus to echo database's output after executing the PL/SQL block.

* The DBMS_OUTPUT.PUT_LINE is called a procedure whose main task is to output a string on the screen.

* To run a PL-SQL code we use / .