Problems of SQL

1) SQL does not have any procedural capabilities.

2) Each time a SQL statement is executed, a call is made to oracle engine's resources. This adds to the traffic to the network thereby decreasing the speed of data processing.

3) SQL has no facility for programmed handling of errors that arise during manipulation of data.

PL/ SQL

PL/SQL which is a **block-structured** language. PL/SQL is a combination of SQL along with the procedural features of programming languages.

PL/ SQL engine resides in oracle engine. The Oracle engine can process not only single SQL statements but also entire PL/SQL blocks.

PL/ SQL Block Structure

Declare Section

Memory Variables and other oracle objects can be declared here.

Begin Section

It contains all SQL executable statements and PL/SQL executable statements.

Exception Section

It deals with exception handling statements.

End Section

It marks the end of a transaction.

PL/ SQL Display

```
DBMS_OUTPUT. PUT_LINE (‘ Hello World ‘);
```

PL/ SQL statement for displaying output.

PL/ SQL Display

DBMS_OUTPUT

A package that includes a number of procedures and functions that accumulate information in a buffer so that it can be retrieved later.

PUT_LINE

Puts a piece of information in the package buffer for displaying a message in form of a string.

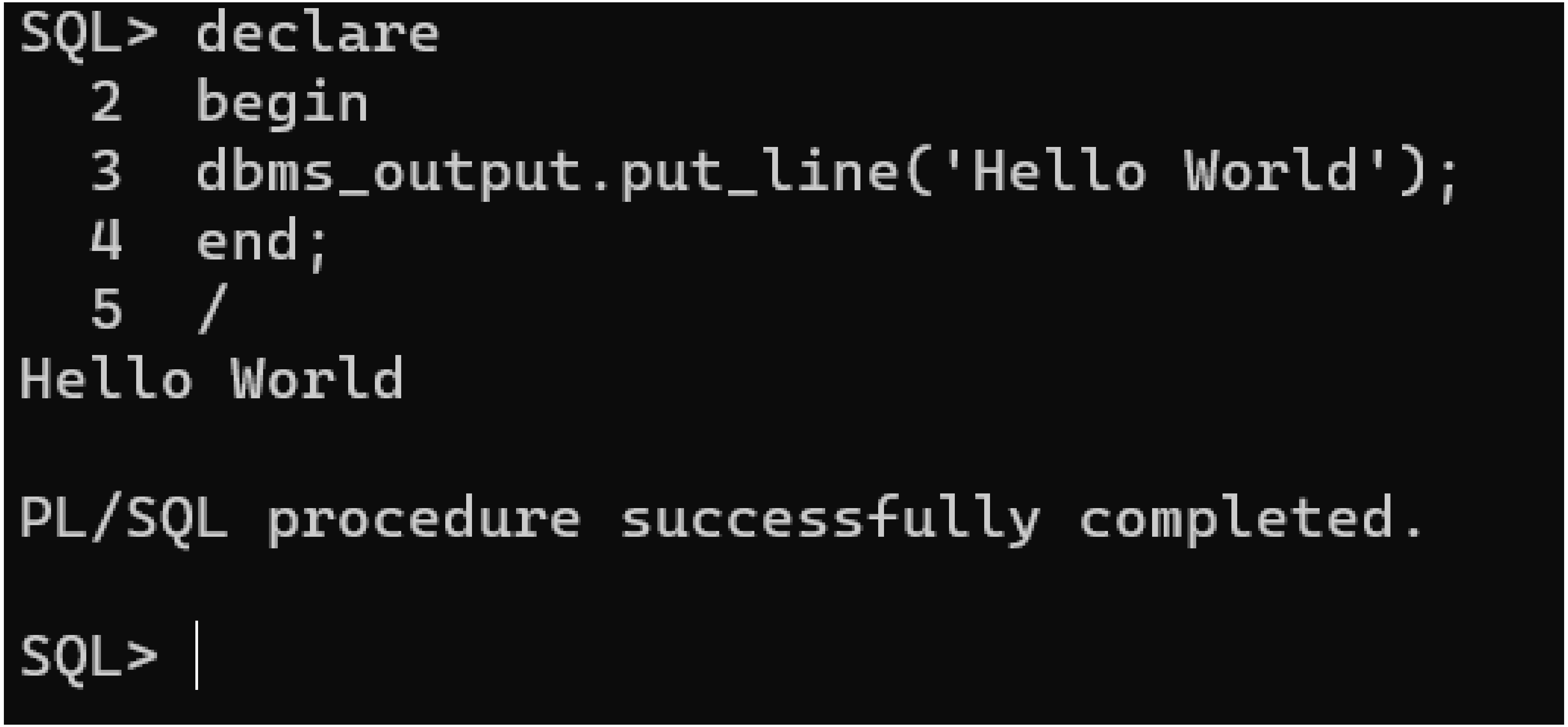
PL/ SQL Display

In SQL environment,

Following statement must be written first to display anything.

```
SET SERVEROUTPUT ON
```

PL/ SQL

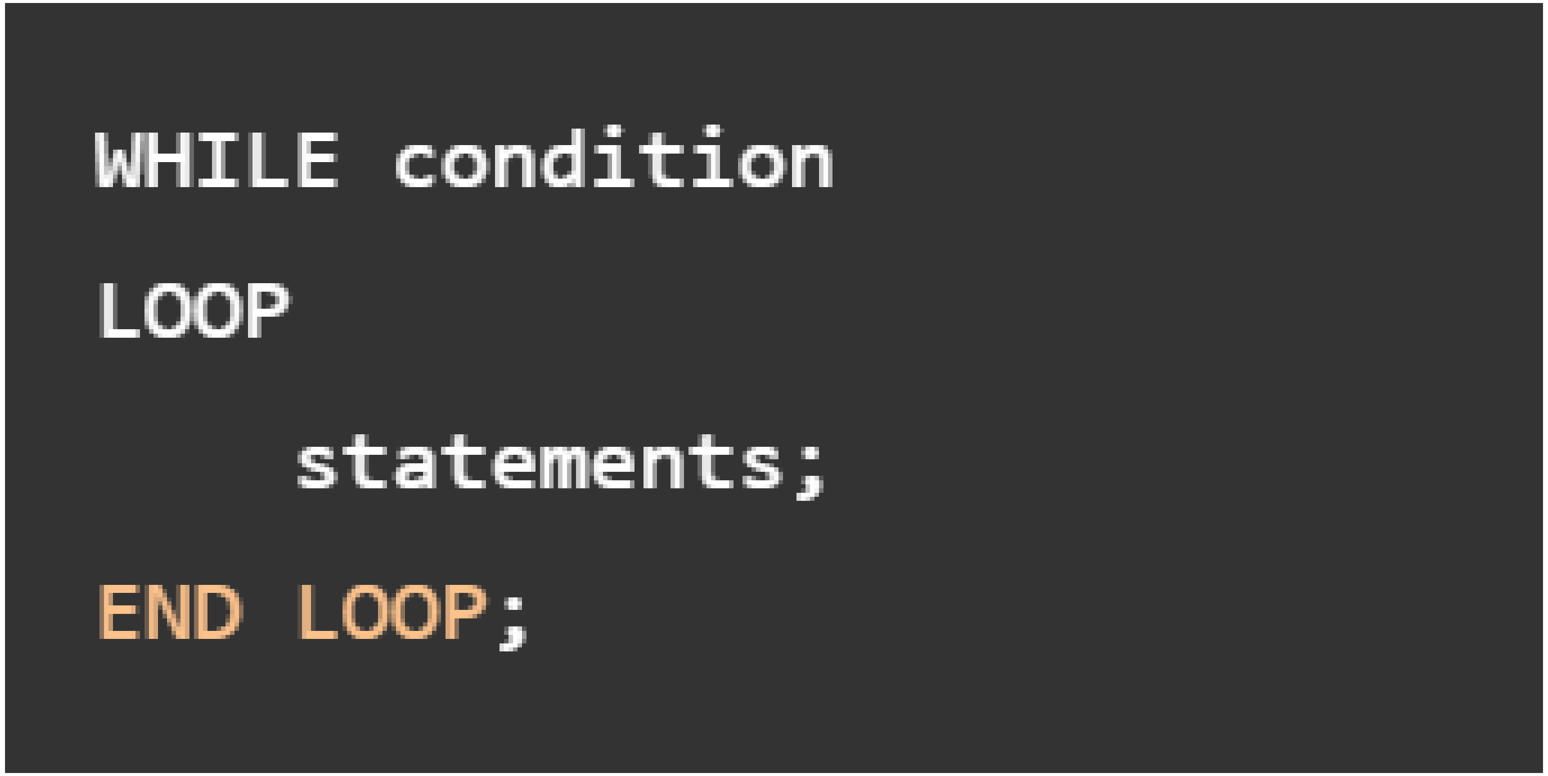


```
SQL> declare
  2  begin
  3  dbms_output.put_line('Hello World');
  4  end;
  5  /
Hello World

PL/SQL procedure successfully completed.

SQL>
```

Conditional in PL/ SQL

```
declare
n number;
begin
n:=&n;
if mod(n,2)=0 then
        dbms_output.put_line('Number is Even');
else
        dbms_output.put_line('Number is odd');
end if;
end;
/
```

while loop IN PL/SQL



```
WHILE condition
LOOP
    statements;
END LOOP;
```

for loop IN PL/SQL

```
DECLARE
  i number(1);
  j number(1);
BEGIN

  FOR i IN 1..5 LOOP
    FOR j IN 1..5 LOOP
      dbms_output.put_line('i is: '| | i | | ' and j is: ' | | j);
    END loop inner_loop;
  END loop outer_loop;
END;
/
```