SE IT	Roll number :
Experiment no. : 7	Date of Implementation :
Aim : To implement PL/pgSQL	
Tool Used : PostgreSQL	
Related Course outcome : At the end of the course, Student should be able to: 4. Write queries in SQL to retrieve any type of information from a data base	

Rubrics for assessment of Experiment:

Indicator	Poor	Average	Good
Timeliness Maintains Experiment deadline (3)	Experiment not done (0)	One or More than One week late (1-2)	Maintains deadline (3)
Completeness and neatness Complete all parts of Experiment(3)	N/A	< 80% complete (1-2)	100% complete (3)
Originality Extent of plagiarism(2)	Copied it from someone else(0)	At least try to implement but could not succeed (1)	Implemented (2)
Knowledge In depth knowledge of the Experiment(2)	Unable to answer any questions(0)	Unable to answer few questions (1)	Able to answer all questions (2)

Assessment Marks :

Timeliness	
Completeness and neatness	
Originality	
Knowledge	
Total	

Total : (Out of 10)

Teacher's Sign :

<table>
<tr><td>EXPERIMENT 7</td><td>PL/pgSQL</td></tr>
<tr><td>Aim</td><td>To implement PL/pgSQL</td></tr>
<tr><td>Tools</td><td>PostgreSQL</td></tr>
<tr><td>Procedure</td><td>PL/pgSQL is a loadable procedural language for the Postgres database system. This package was originally written by Jan Wieck. The design goals of PL/pgSQL were to create a loadable procedural language that can be used to create functions and trigger procedures, adds control structures to the SQL language.
Structure of PL/pgSQL
PL/pgSQL is a block-structured language. The complete text of a function definition must be a block. A block is defined as:
[<<label>>]
[DECLARE Declarations]
BEGIN
statements
END [label];
Each declaration and each statement within a block is terminated by a semicolon. A block that appears within another block must have a semicolon after END , as shown above; however the final END that concludes a function body does not require a semicolon
<table>
<tr><td>IF boolean-expression THEN
statements
END IF;</td><td>IF boolean-expression
THEN
statements
ELSE statements
END IF;</td></tr>
</table>
<table>
<tr><td>WHILE boolean-expression
LOOP
statements
END LOOP [label];</td><td>FOR name IN [REVERSE] expression..expression
[BY expression] LOOP
statements
END LOOP [label];
FOR i IN 1..10 LOOP
-- i will take on the values 1,2,3,4,5,6,7,8,9,10 within the loop
END LOOP;
FOR i IN REVERSE 10..1 LOOP
-- i will take on the values 10,9,8,7,6,5,4,3,2,1 within the loop
END LOOP;
FOR i IN REVERSE 10..1 BY 2 LOOP
-- i will take on the values 10,8,6,4,2 within the loop
END LOOP;</td></tr>
</table></td></tr>
<tr><td>Procedure</td><td>1. Write a block to display sum of digits of a three digit number
2. Write a block to display square of 1 to 10
3. Write a block to display Fibonacci series upto 8th term (start with 0,1)</td></tr>
<tr><td>Post Lab Questions:</td><td>1. Give advantages of PgSQL
2. Explain data types of PgSQL</td></tr>
</table>