Experiment 8 PL/SQL

1. **Write a PL/SQL program which processes a bank transaction. Before allowing you to withdraw**
   $500 from account 3, it makes sure the account has sufficient funds to cover the withdrawal. Ifthe funds are available, the program debits the account. Otherwise, the program prints a message "insufficient funds".

ACCOUNTS TABLE

ACCOUNT ID	ACCOUNT TYPE	ACC BALANCE	CC_HOLDER NAME
1	SAVINGS	1500	JAMES
2	CURRENT	300	JOHN
3	SAVINGS	3000	SMITH
4	SAVINGS	4000	ADAMS
5	CURRENT	5000	FORD

```
SQL> DECLARE
  2    acct_balance NUMBER(11,2);
  3    acct         CONSTANT NUMBER(4) := 3;
  4    debit_amt    CONSTANT NUMBER(5,2) := 500.00;
  5  BEGIN
  6    SELECT bal INTO acct_balance FROM accounts
  7      WHERE id = acct
  8      FOR UPDATE OF bal;
  9    IF acct_balance >= debit_amt THEN
 10      UPDATE accounts SET bal = bal - debit_amt
 11        WHERE id = acct;
 12    ELSE
 13      DBMS_OUTPUT.PUT_LINE('insufficient funds');
 14    END IF;
 15    COMMIT;
 16  END;
 17  /
```

```
SQL> spool ex8.lst
SQL> select * from accounts;

        ID TYPE            BAL ACC_H
---------- -------- ---------- -----
         1 savings        1500 James
         2 current         300 John
         3 savings        3000 Smith
         4 savings        4000 Adams
         5 current        5000 Ford

SQL> DECLARE
  2     acct_balance NUMBER(11,2);
  3     acct         CONSTANT NUMBER(4) := 3;
  4     debit_amt    CONSTANT NUMBER(5,2) := 500.00;
  5  BEGIN
  6     SELECT bal INTO acct_balance FROM accounts
  7        WHERE id = acct
  8        FOR UPDATE OF bal;
  9     IF acct_balance >= debit_amt THEN
 10        UPDATE accounts SET bal = bal - debit_amt
 11           WHERE id = acct;
 12     ELSE
 13        DBMS_OUTPUT.PUT_LINE('insufficient funds');
 14     END IF;
 15     COMMIT;
 16  END;
 17  /

PL/SQL procedure successfully completed.

SQL> select * from accounts;

        ID TYPE            BAL ACC_H
---------- -------- ---------- -----
         1 savings        1500 James
         2 current         300 John
         3 savings        2500 Smith
         4 savings        4000 Adams
         5 current        5000 Ford
```

2. Write a PL/SQL program for finding the area of square, circle, and rectangle using switch case.

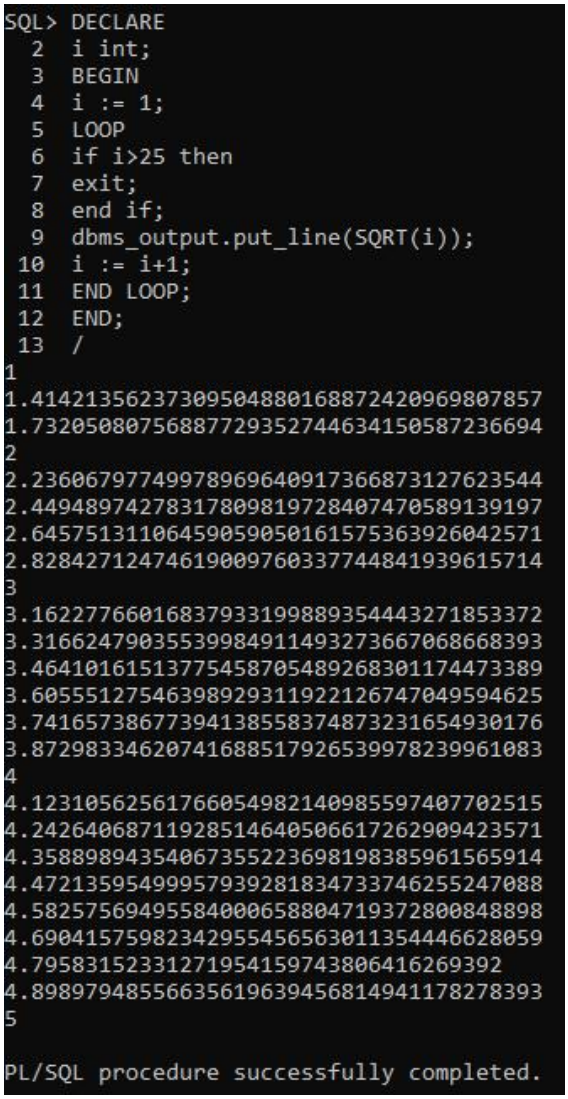
```
SQL> DECLARE
  2     L NUMBER(4,2) := 3;
  3     B NUMBER(4,2) := 7;
  4     R NUMBER(1) := 5;
  5     squarearea NUMBER(4,2);
  6     rectarea NUMBER(4,2);
  7     circlearea NUMBER(6,2);
  8     pi CONSTANT NUMBER(3,2) := 3.14;
  9  BEGIN
 10     squarearea := L * L;
 11     rectarea := L * B;
 12     circlearea := pi * R * R;
 13     dbms_output.Put_line('Area of square is ' || squarearea);
 14     dbms_output.Put_line('Area of rectangle is ' || rectarea);
 15     dbms_output.Put_line('Area of circle is ' || circlearea);
 16  END;
 17  /
```

```
SQL> DECLARE
  2     L NUMBER(4,2) := 3;
  3     B NUMBER(4,2) := 7;
  4     R NUMBER(1) := 5;
  5     squarearea NUMBER(4,2);
  6     rectarea NUMBER(4,2);
  7     circlearea NUMBER(6,2);
  8     pi CONSTANT NUMBER(3,2) := 3.14;
  9  BEGIN
 10     squarearea := L * L;
 11     rectarea := L * B;
 12     circlearea := pi * R * R;
 13     dbms_output.Put_line('Area of square is ' || squarearea);
 14     dbms_output.Put_line('Area of rectangle is ' || rectarea);
 15     dbms_output.Put_line('Area of circle is ' || circlearea);
 16  END;
 17  /
Area of square is 9
Area of rectangle is 21
Area of circle is 78.5

PL/SQL procedure successfully completed.
```

3. Write a PL/SQL program for finding the square roots of 1 to 25 using for loop.

```
SQL> DECLARE
  2  i int;
  3  BEGIN
  4  i := 1;
  5  LOOP
  6  if i>25 then
  7  exit;
  8  end if;
  9  dbms_output.put_line(SQRT(i));
 10  i := i+1;
 11  END LOOP;
 12  END;
 13  /
```



```
SQL> DECLARE
  2  i int;
  3  BEGIN
  4  i := 1;
  5  LOOP
  6  if i>25 then
  7  exit;
  8  end if;
  9  dbms_output.put_line(SQRT(i));
 10  i := i+1;
 11  END LOOP;
 12  END;
 13  /
1
1.41421356237309504880168872420969807857
1.73205080756887729352744634150587236694
2
2.23606797749978969640917366873127623544
2.44948974278317809819728407470589139197
2.64575131106459059050161575363926042571
2.82842712474619009760337744841939615714
3
3.16227766016837933199889354443271853372
3.31662479035539984911493273667068668393
3.46410161513775458705489268301174473389
3.60555127546398929311922126747049594625
3.74165738677394138558374873231654930176
3.87298334620741688517926539978239961083
4
4.12310562561766054982140985597407702515
4.24264068711928514640506617262909423571
4.35889894354067355223698198385961565914
4.47213595499957939281834733746255247088
4.58257569495584000658804719372800848898
4.69041575982342955456563011354446628059
4.79583152331271954159743806416269392
4.89897948556635619639456814941178278393
5

PL/SQL procedure successfully completed.
```