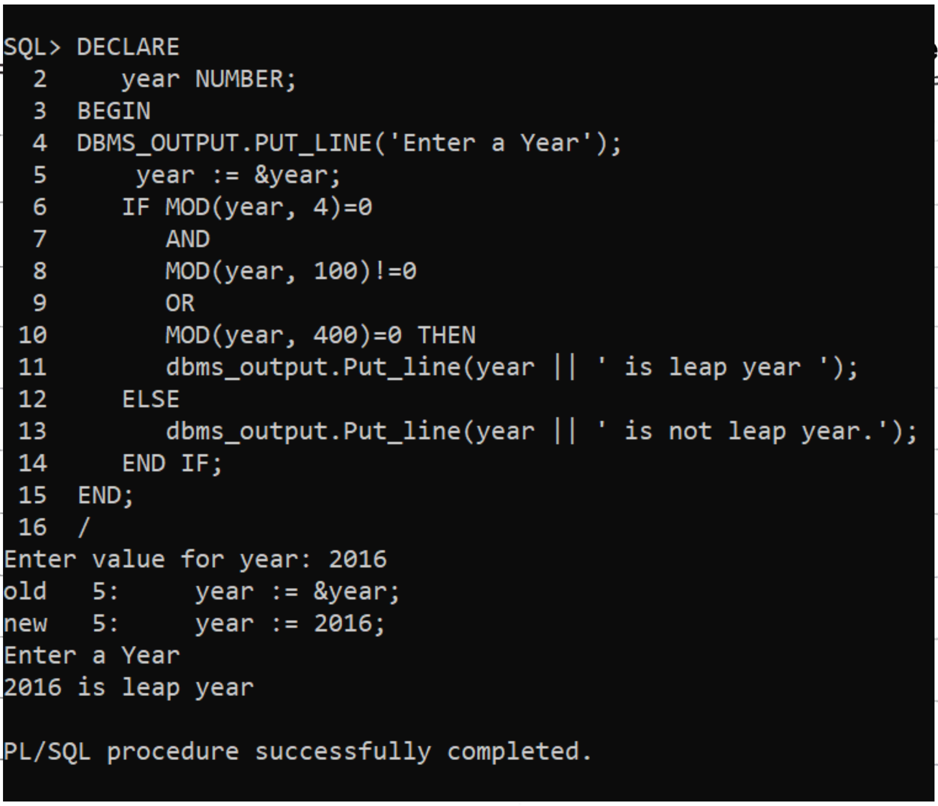
Date ....

Name Nikhil Gupta
Roll No. 20051523
Batch CSE - 6
Branch CSE
Subject DBMS

Assignment - 9

Ques 1 Write a PL/SQL block to find out if a year is leap year or not

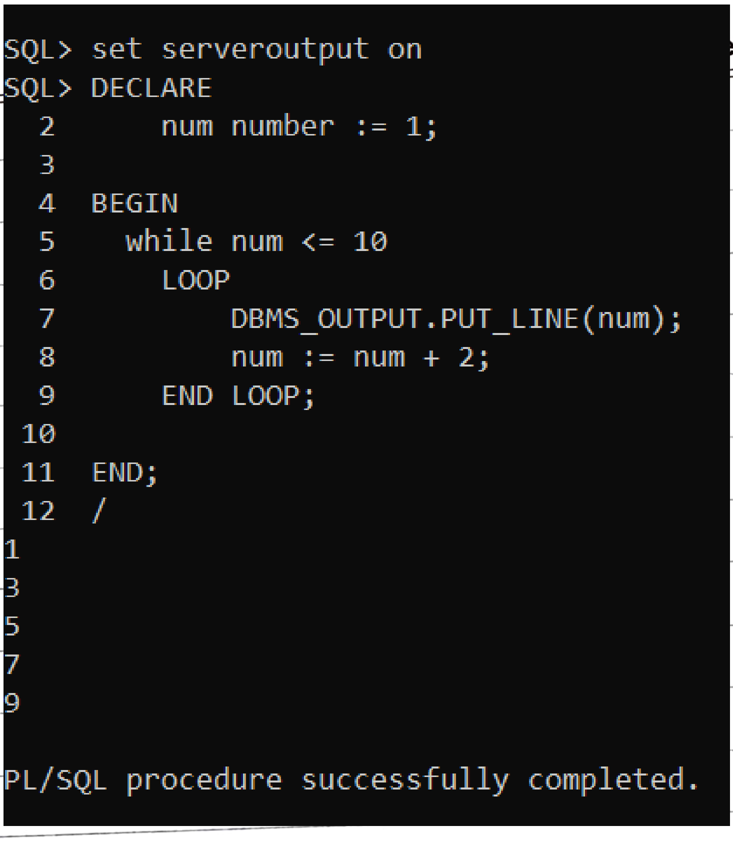
```
DECLARE
        year Number;

BEGIN

    DBMS_OUTPUT.PUT_LINE('Enter a Year');

          year := &year;

    IF mod(year, 4) = 0

    AND
       MOD(year, 100) != 0
    OR
       MOD(year, 400) = 0
   THEN
     DBMS_OUTPUT.PUT_LINE(year || 'is Leap year');

    ELSE
       DBMS_OUTPUT.PUT_LINE(year || 'is not a leap year');
   END IF;
 END;
```



```
SQL> DECLARE
        year NUMBER;
     BEGIN
     DBMS_OUTPUT.PUT_LINE('Enter a Year');
         year := &year;
        IF MOD(year, 4)=0
           AND
           MOD(year, 100)!=0
           OR
           MOD(year, 400)=0 THEN
           dbms_output.Put_line(year || ' is leap year ');
        ELSE
           dbms_output.Put_line(year || ' is not leap year.');
        END IF;
     END;
     /
Enter value for year: 2016
old   5:      year := &year;
new   5:      year := 2016;
Enter a Year
2016 is leap year

PL/SQL procedure successfully completed.
```

Ques2 Write a PL/SQL block to print odd numbers between 1 & 10 using Loop.

```
DECLARE
        num number := 1;

BEGIN
      while num <= 10 LOOP

             DBMS_OUTPUT.PUT_LINE(num);
             num := num + 2;

      END LOOP;
END;
```

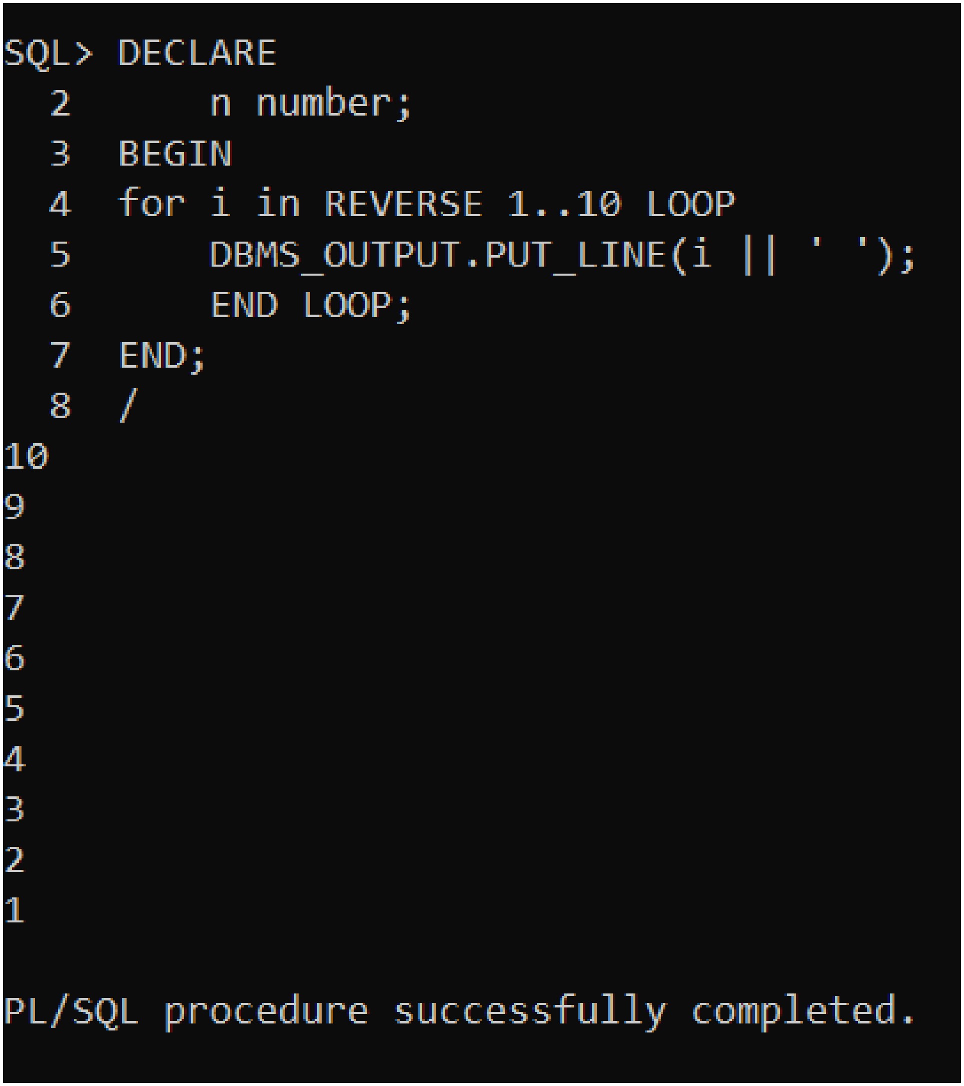


```
SQL> set serveroutput on
SQL> DECLARE
         num number := 1;

     BEGIN
       while num <= 10
         LOOP
             DBMS_OUTPUT.PUT_LINE(num);
             num := num + 2;
         END LOOP;

     END;
     /
1
3
5
7
9

PL/SQL procedure successfully completed.
```

Ques 3 Using a for Loop, print the values 10 to 1 in reverse order.

~~DELA~~

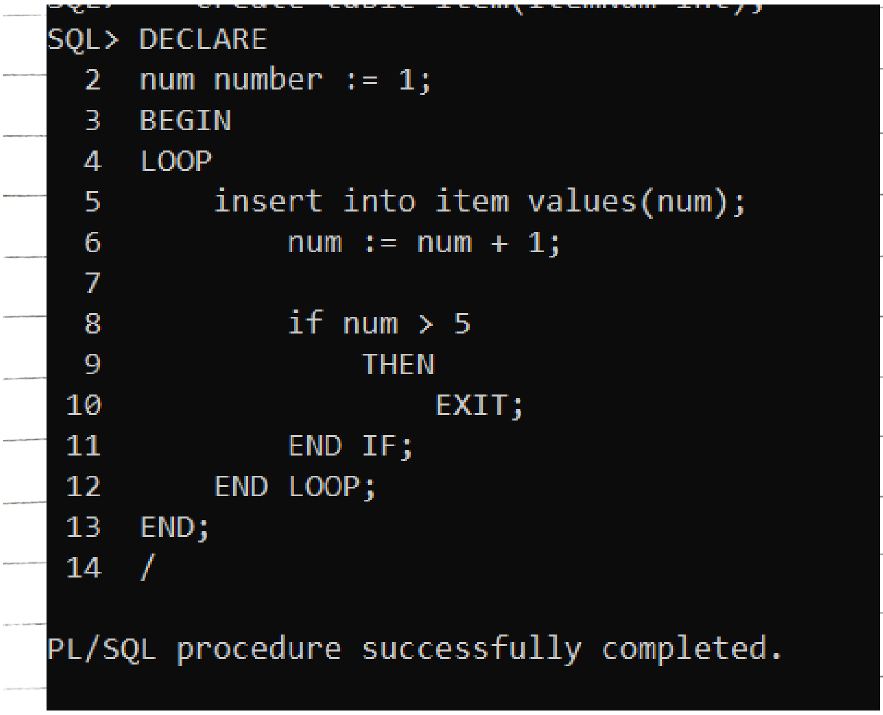
```
DECLARE
      n number;

BEGIN
   for i in REVERSE 1.. 10 LOOP
         DBMS_OUTPUT. PUT_LINE(i || ' ');
   END LOOP;
END;
```



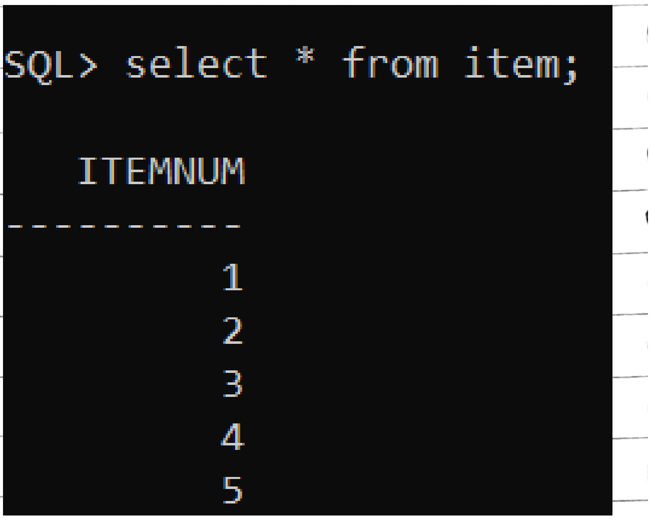
```
SQL> DECLARE
  2      n number;
  3  BEGIN
  4  for i in REVERSE 1..10 LOOP
  5      DBMS_OUTPUT.PUT_LINE(i || ' ');
  6      END LOOP;
  7  END;
  8  /
10
9
8
7
6
5
4
3
2
1

PL/SQL procedure successfully completed.
```

Ques 4 Create a table called item with one column Item Num with NUMBER type. Write PL/SQL program to insert values of 1 to 5 for item num.

```
create table item(ItemNum int);

DECLARE
    num NUMBER := 1;
BEGIN
    LOOP
        INSERT INTO ITEM VALUES(num);
            num := num + 1;

            IF num > 5
             THEN
                 EXIT;
            END IF;
        END LOOP;
    END;
```



```
SQL>    create table item(itemnum int);
SQL> DECLARE
  num number := 1;
  BEGIN
  LOOP
      insert into item values(num);
          num := num + 1;

          if num > 5
              THEN
                  EXIT;
          END IF;
      END LOOP;
  END;
  /

PL/SQL procedure successfully completed.
```

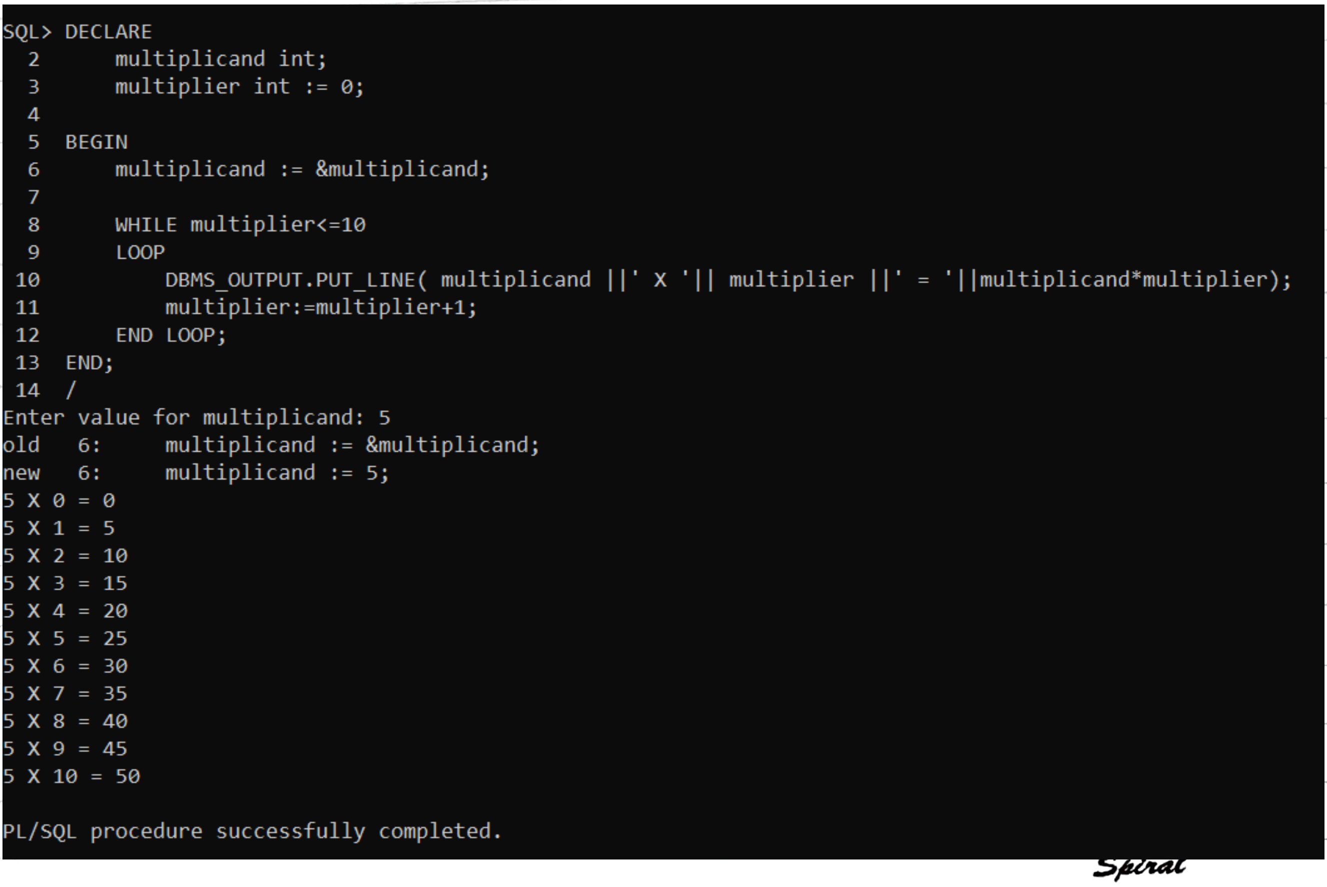


```
SQL> select * from item;

   ITEMNUM
----------
         1
         2
         3
         4
         5
```

Date ....

Ques 5 Input a number with a substitution value & then print its multiplication table using while Loop

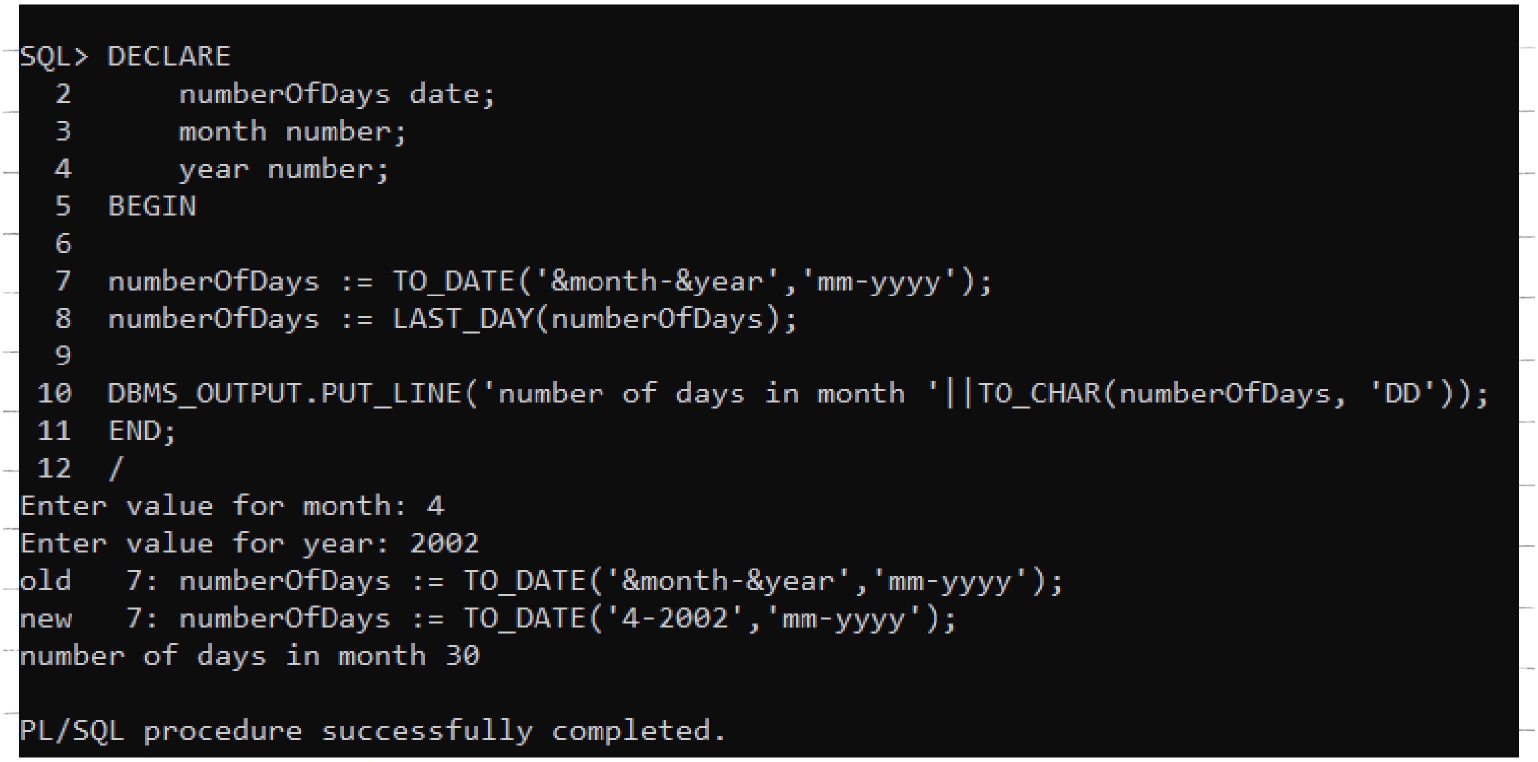
```
DECLARE
    multiplicand int;
    multiplier int := 0;

BEGIN
  multiplicand := &multiplicand ;

   WHILE multiplier <=10
   LOOP
     DBMS_OUTPUT.PUT_LINE(multiplicand ||' X '||
       multiplier ||' = ' || multiplicand * multiplier);
   END LOOP;
END;
```



```
SQL> DECLARE
  2      multiplicand int;
  3      multiplier int := 0;
  4
  5  BEGIN
  6      multiplicand := &multiplicand;
  7
  8      WHILE multiplier<=10
  9      LOOP
 10          DBMS_OUTPUT.PUT_LINE( multiplicand ||' X '|| multiplier ||' = '||multiplicand*multiplier);
 11          multiplier:=multiplier+1;
 12      END LOOP;
 13  END;
 14  /
Enter value for multiplicand: 5
old   6:      multiplicand := &multiplicand;
new   6:      multiplicand := 5;
5 X 0 = 0
5 X 1 = 5
5 X 2 = 10
5 X 3 = 15
5 X 4 = 20
5 X 5 = 25
5 X 6 = 30
5 X 7 = 35
5 X 8 = 40
5 X 9 = 45
5 X 10 = 50

PL/SQL procedure successfully completed.
```

Ques 6 Input a month number between 1 & 12 & a 4 digit year, & print number of days in month

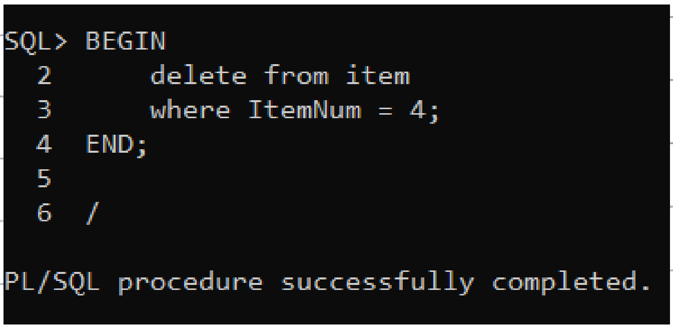
```
 DECLARE
        numberOfDays date;
        month number;
        year number;

BEGIN
     number ofDays := To_Date('&month-&year', 'mm-yyyy');
     numberOfDays := LAST_DAY(numberOfDays);

     DBMS_OUTPUT.PUT_LINE('number of Days in month '||
                   TO_CHAR(numberofDays, 'DD'));
END;
```



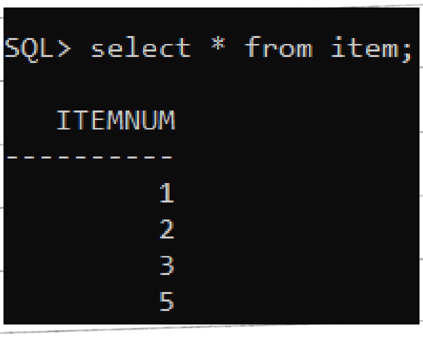
```
SQL> DECLARE
  2      numberOfDays date;
  3      month number;
  4      year number;
  5  BEGIN
  6
  7  numberOfDays := TO_DATE('&month-&year','mm-yyyy');
  8  numberOfDays := LAST_DAY(numberOfDays);
  9
 10  DBMS_OUTPUT.PUT_LINE('number of days in month '||TO_CHAR(numberOfDays, 'DD'));
 11  END;
 12  /
Enter value for month: 4
Enter value for year: 2002
old   7: numberOfDays := TO_DATE('&month-&year','mm-yyyy');
new   7: numberOfDays := TO_DATE('4-2002','mm-yyyy');
number of days in month 30

PL/SQL procedure successfully completed.
```

Ques 7 Use a PL/SQL to delete item 4 from item tabl

```
BEGIN
    delete from item
    where itemNum = 4;
END;
```



```
SQL> BEGIN
        delete from item
        where ItemNum = 4;
     END;

     /

PL/SQL procedure successfully completed.
```

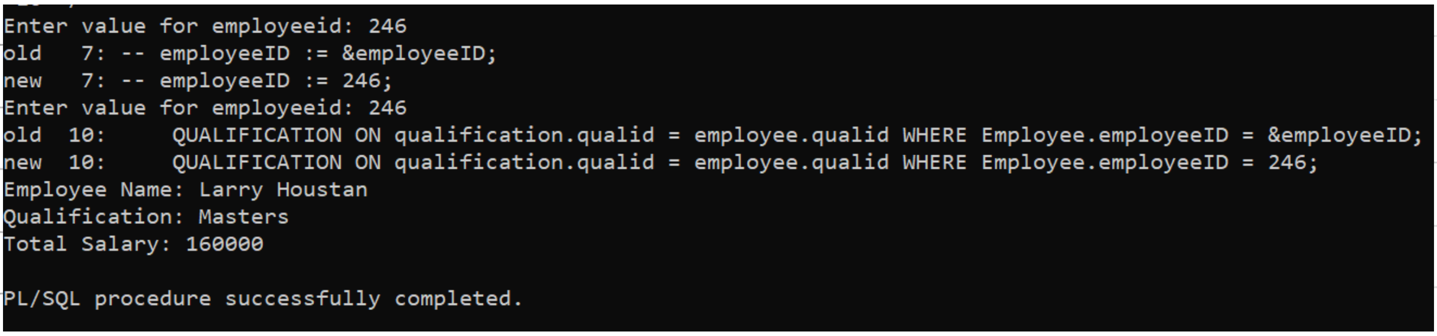


```
SQL> select * from item;

   ITEMNUM
----------
         1
         2
         3
         5
```

Ques 8 Write a PL/SQL block to ask a user to input a valid employee Id. Retrieve employee name, qualifica description, salary + commission.

```
DECLARE
    v_emp employee%ROWTYPE;
    v_qual qualification%ROWTYPE;
    employeeID number;

BEGIN
  select employee.fname, employee.Lname, employee.salary,
  qualification.qualdesc, employee.commission INTO
   v_emp.fname, v_emp.lname, v_emp.salary, v_qual.
   desc, v_emp.commission from employee INNER JOIN
  qualification ON qualification.qualid = employee.qualid
  where Employee.employeeID = &employeeID;

  DBMS_OUTPUT.PUT_LINE('Employee Name:' || v_emp.fname ||
          ' ' || v_emp.lname);
```

```
DBMS_OUTPUT.PUT_LINE('Qualification:   '|| v_qual.desc)

DBMS_OUTPUT.PUTLINE('Total Salary: '|| (v_emp.salary +
                    v_emp.commission));

END;
```

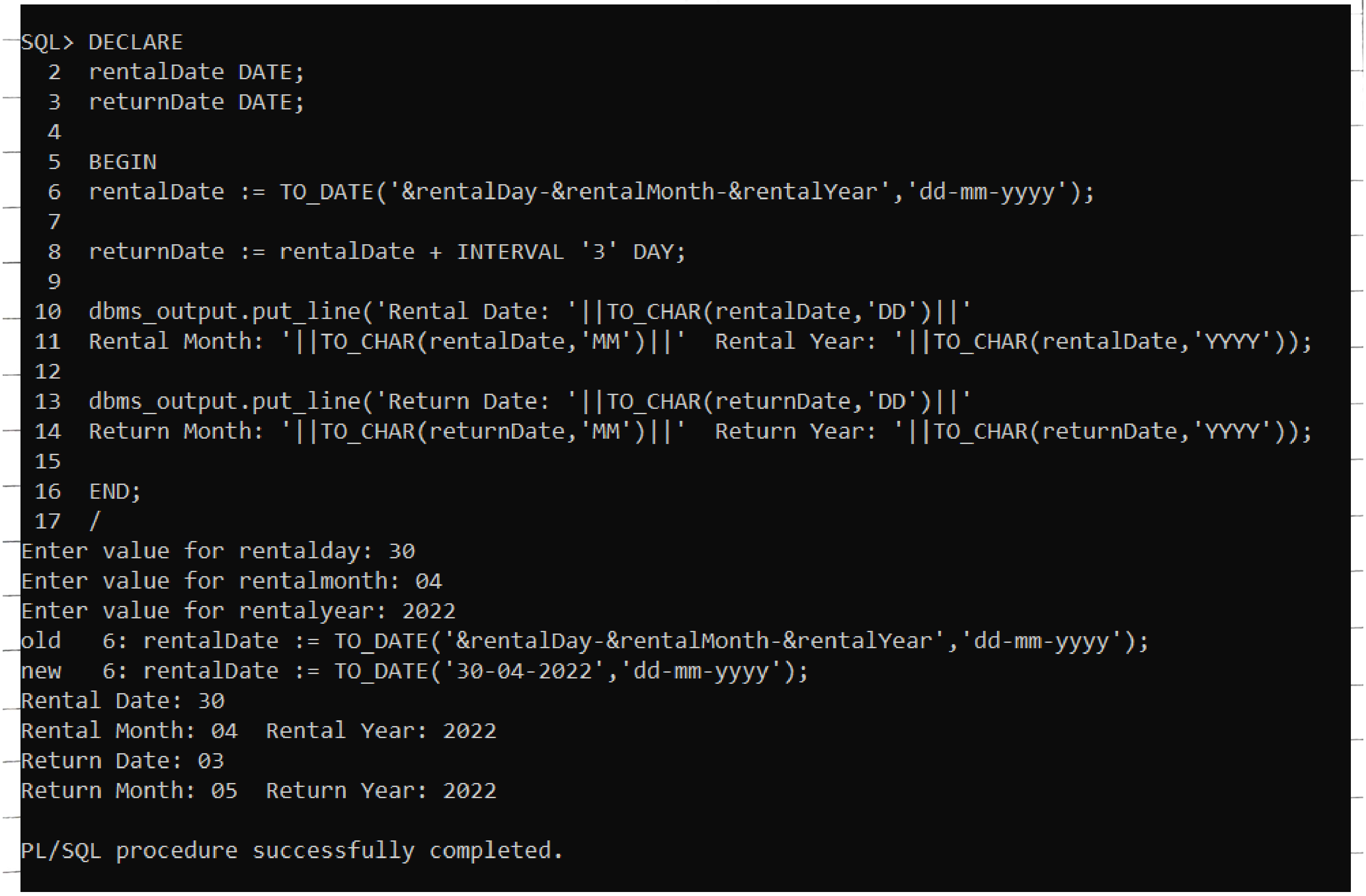


```
Enter value for employeeid: 246
old   7: -- employeeID := &employeeID;
new   7: -- employeeID := 246;
Enter value for employeeid: 246
old  10:     QUALIFICATION ON qualification.qualid = employee.qualid WHERE Employee.employeeID = &employeeID;
new  10:     QUALIFICATION ON qualification.qualid = employee.qualid WHERE Employee.employeeID = 246;
Employee Name: Larry Houstan
Qualification: Masters
Total Salary: 160000

PL/SQL procedure successfully completed.
```

Ques 9 You went to a video store & rented a DVD that is due in 3 days from the rental date. Input rental date, rental month & rental year. Calculate & print return date, return month & return year

```
DECLARE
    rentalDate Date;
    returnDate Date;

BEGIN
   rentalDate := To_Date('&rentalDay-&rentalMonth-&rentalYea
           , 'dd-mm-YYYY');

  returnDate := rentalDate + INTERVAL '3' DAY;
```

```
dbms_output.put_Line('Rental Date: '||To_Char(rentalDate,
    'DD')||' RentalMonth: '||To_Char(rentalDate, 'MM')
|| 'Rental Year: '||To_Char(rentalDate, 'yyyy'));

dbms_output.put_Line('Return Date: '||To_Char(returnDate,
   'DD')|| To_Char(returnDate, 'MM') ||' ReturnYear: '||
    To_Char(returnDate, 'yyyy'));

END;
```



```
SQL> DECLARE
  2  rentalDate DATE;
  3  returnDate DATE;
  4
  5  BEGIN
  6  rentalDate := TO_DATE('&rentalDay-&rentalMonth-&rentalYear','dd-mm-yyyy');
  7
  8  returnDate := rentalDate + INTERVAL '3' DAY;
  9
 10  dbms_output.put_line('Rental Date: '||TO_CHAR(rentalDate,'DD')||'
 11  Rental Month: '||TO_CHAR(rentalDate,'MM')||'  Rental Year: '||TO_CHAR(rentalDate,'YYYY'));
 12
 13  dbms_output.put_line('Return Date: '||TO_CHAR(returnDate,'DD')||'
 14  Return Month: '||TO_CHAR(returnDate,'MM')||'  Return Year: '||TO_CHAR(returnDate,'YYYY'));
 15
 16  END;
 17  /
Enter value for rentalday: 30
Enter value for rentalmonth: 04
Enter value for rentalyear: 2022
old   6: rentalDate := TO_DATE('&rentalDay-&rentalMonth-&rentalYear','dd-mm-yyyy');
new   6: rentalDate := TO_DATE('30-04-2022','dd-mm-yyyy');
Rental Date: 30
Rental Month: 04  Rental Year: 2022
Return Date: 03
Return Month: 05  Return Year: 2022

PL/SQL procedure successfully completed.
```