Write a PL/SQL block to find out if a year is a leap year. A leap year is divisibleby 4, but not divisible by 100, or it is divisible by 400. For example, 2000and2004 are leap years, but 1900 and 2001 are not leap years. (Hint: The functionMOD (n, d) divides n by d and returns the integer remainder fromthe operation)

DECLARE

year NUMBER := :year; -- Input the year here

is\_leap\_year BOOLEAN := FALSE;

BEGIN

IF (year MOD 4 = 0 AND year MOD 100 != 0) OR (year MOD 400 = 0) THEN

is\_leap\_year := TRUE;

END IF;

IF is\_leap\_year THEN

DBMS\_OUTPUT.PUT\_LINE(year || ' is a leap year.');

ELSE

DBMS\_OUTPUT.PUT\_LINE(year || ' is not a leap year.');

END IF;

END;

/

Write a PL/SQL block to print all odd numbers between 1 and 10 usingabasicloop.

declare

n number;

begin

n := 1;

while n<=10

loop

dbms\_output.put\_line(n);

n := n+2;

end loop;

end;

/

Using a FOR loop, print the values 10 to 1 in reverse order.

DECLARE

VAR1 NUMBER;

BEGIN

FOR VAR1 IN REVERSE 1..10

LOOP

DBMS\_OUTPUT.PUT\_LINE (VAR1);

END LOOP;

END;

/

Create a table called ITEM with one column ItemNumwith NUMBERtype. Write a PL/SQL program to insert values of 1 to 5 for ItemNum.

CREATE TABLE ITEM

(

ItemNum NUMBER

);

DECLARE

n number:=1;

BEGIN

insert into item values (1);

insert into item values (2);

insert into item values (3);

insert into item values (4);

insert into item values (5);

END ;

/

Input a number with a substitution variable and then print its multiplicationtableusing a WHILE loop.

DECLARE

n number(2);

i number(2);

BEGIN

n := :n;

for i in 1..10

loop

dbms\_output.put\_line(n|| '\*' ||i|| '=' || n\*i);

end loop;

END;

/

Use a PL/SQL block to delete item number 4 from the ITEMtable

DECLARE

BEGIN

delete from item where itemnum=4;

END;

/

Write a PL/SQL block to ask a user to input a valid employee Id. Retrieveemployees name, qualification description, salary and commission. Print name, qualification and sum of salary and commission

DECLARE

ID NUMBER(4);

NAME VARCHAR2(20);

QUAL VARCHAR2(10);

SALARY NUMBER(6);

BEGIN

ID:= :ID;

SELECT INITCAP(F\_NAME) || ',' || INITCAP(L\_NAME) ,

QUAL\_DESCRIPTION , SALARY + NVL(COMMISSION,0) INTO NAME , QUAL , SALARY

FROM EMPLOYEE JOIN QUALIFICATION USING(QUALID) WHERE EMPLOYEEID=ID;

DBMS\_OUTPUT.PUT\_LINE( ' Name : ' || NAME );

DBMS\_OUTPUT.PUT\_LINE( ' Qualification : ' || QUAL );

DBMS\_OUTPUT.PUT\_LINE( ' Total salary : ' || SALARY );

END;

/