Q1. Write a PL/SQL code to print Today is fall on weekend or weekdays using if else statement.

```
DECLARE
   v_day VARCHAR2(15);
BEGIN
   SELECT TO_CHAR(SYSDATE, 'Day') INTO v_day FROM DUAL;

   IF v_day IN ('Saturday', 'Sunday') THEN
      DBMS_OUTPUT.PUT_LINE('Today is a weekend.');
   ELSE
      DBMS_OUTPUT.PUT_LINE('Today is a weekday.');
   END IF;
END;
/
```

```
Statement processed.
Today is a weekday.
```

Q2. Write a PL/SQL code to check that an inputted a single character is vowel or not .If vowel then display which vowel it is.

```
DECLARE
   v_in_char CHAR(1);
        v_char CHAR(1);
        v_c CHAR(1);
BEGIN
   DBMS_OUTPUT.PUT_LINE('Enter a single character: ');
   v_in_char := UPPER(SUBSTR(TRIM('&1'), 1, 1));

        v_char := 'b';
        v_c := UPPER(SUBSTR(v_char, 1, 1));

   IF v_c IN ('A', 'E', 'I', 'O', 'U') THEN
      DBMS_OUTPUT.PUT_LINE(v_char || ' is a vowel.');
   ELSE
      DBMS_OUTPUT.PUT_LINE(v_char || ' is not a vowel.');
   END IF;
END;
/
```

```
Statement processed.
Enter a single character:
b is not a vowel.
```

Q3. Write a PL/SQL code block to find out the sum of first twenty natural numbers (1+2+3+4+5+6+7+8+9+10+-----+20 this series).

```
DECLARE
   v_sum NUMBER := 0;
   v_count NUMBER := 1;
BEGIN

   WHILE v_count <= 20 LOOP
      v_sum := v_sum + v_count;
      v_count := v_count+ 1;
   END LOOP;

   DBMS_OUTPUT.PUT_LINE('The sum of the first 20 natural numbers is: ' ||
v_sum);
END;
/
```

```
Statement processed.
The sum of the first 20 natural numbers is: 210
```

Q4. Write a PL/SQL block that will ask for two numbers and one operand (+, -, *, /). Then it will calculate and display the result.

```
DECLARE
   v_num1 NUMBER;
   v_num2 NUMBER;
   v_operator CHAR(1);
   v_result NUMBER;
BEGIN
   DBMS_OUTPUT.PUT_LINE('Enter the first number: ');
   -- v_num1 := &num1;
        v_num1 := 10;
        DBMS_OUTPUT.PUT_LINE('Entered number is: ' || v_num1);

        DBMS_OUTPUT.PUT_LINE('Enter the second number: ');
   -- v_num2 := &num2;
        v_num2 := 3;
        DBMS_OUTPUT.PUT_LINE('Entered number is: ' || v_num2);

        DBMS_OUTPUT.PUT_LINE('Enter the operand (+, -, *, /): ');
   -- v_operator := '&operator';
        v_operator := '/';

   CASE v_operator
      WHEN '+' THEN
         v_result := v_num1 + v_num2;
      WHEN '-' THEN
         v_result := v_num1 - v_num2;
      WHEN '*' THEN
         v_result := v_num1 * v_num2;
      WHEN '/' THEN
         IF v_num2 = 0 THEN
            DBMS_OUTPUT.PUT_LINE('Error: Division by zero.');
         ELSE
            v_result := v_num1 / v_num2;
         END IF;
      ELSE
```

```
            DBMS_OUTPUT.PUT_LINE('Invalid operator.');
    END CASE;

    IF v_operator IN ('+', '-', '*', '/') THEN
        DBMS_OUTPUT.PUT_LINE('Result: ' || v_num1 || ' ' || v_operator || ' ' ||
v_num2 || ' = ' || v_result);
    END IF;
END;
/
```

```
Statement processed.
Enter the first number:
Entered number is: 10
Enter the second number:
Entered number is: 3
Enter the operand (+, -, *, /):
Result: 10 / 3 = 3.333333333333333333333333333333333333
```

Q5. Write a PL/SQL code block to display a number in a reverse way.

```
DECLARE
   v_number NUMBER := 12345;
   v_dummy NUMBER;
   v_reverse NUMBER := 0;
BEGIN
   v_dummy := v_number;
   WHILE v_dummy > 0 LOOP
      v_reverse := v_reverse * 10 + MOD(v_dummy, 10);
      v_dummy := TRUNC(v_dummy / 10);
   END LOOP;

   DBMS_OUTPUT.PUT_LINE('Original Number: ' || v_number);
   DBMS_OUTPUT.PUT_LINE('Reversed Number: ' || v_reverse);
END;
/
```

```
Statement processed.
Original Number: 12345
Reversed Number: 54321
```

Q6. Write a PL/SQL block to display the dates of this month which are Tuesday

```
DECLARE
   v_start_date DATE;
   v_end_date DATE;
BEGIN
   v_start_date := TRUNC(SYSDATE, 'MM');

   v_end_date := LAST_DAY(SYSDATE);

   WHILE v_start_date <= v_end_date LOOP
      IF TO_CHAR(v_start_date, 'D') = '3' THEN
         DBMS_OUTPUT.PUT_LINE('Tuesday: ' || TO_CHAR(v_start_date,
'YYYY-MM-DD'));
      END IF;
      v_start_date := v_start_date + 1;
   END LOOP;
END;
/
```

```
Statement processed.
Tuesday: 2023-10-03
Tuesday: 2023-10-10
Tuesday: 2023-10-17
Tuesday: 2023-10-24
Tuesday: 2023-10-31
```

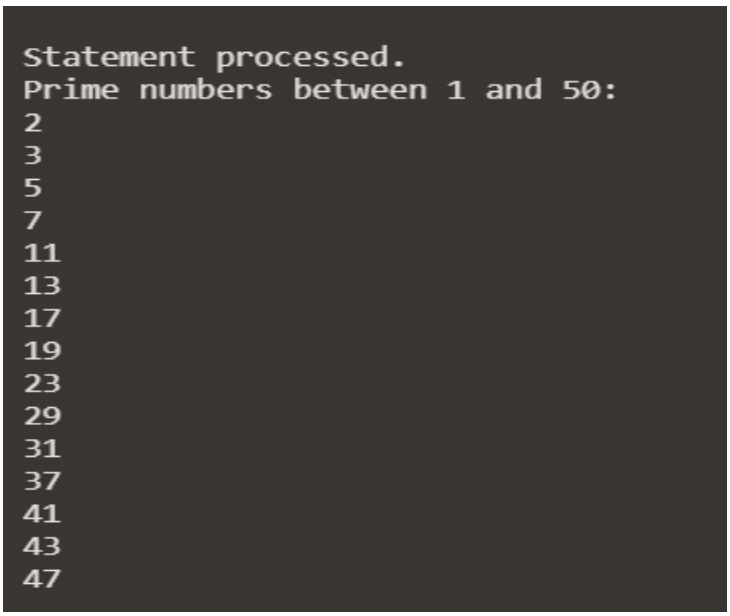
Q7. Write a program in PL/SQL to print the prime numbers between 1 to 50.

```
DECLARE
   v_number NUMBER;
   v_is_prime BOOLEAN;
BEGIN
   DBMS_OUTPUT.PUT_LINE('Prime numbers between 1 and 50:');

   FOR v_number IN 2..50 LOOP
      v_is_prime := TRUE;

      FOR i IN 2..FLOOR(SQRT(v_number)) LOOP
         IF MOD(v_number, i) = 0 THEN
            v_is_prime := FALSE;
            EXIT;
         END IF;
      END LOOP;

      IF v_is_prime THEN
         DBMS_OUTPUT.PUT_LINE(v_number);
      END IF;
   END LOOP;
END;
/
```



Q8. Write a program in PL/SQL to print the sum of digits of a number [eg: 635=14].

```
DECLARE
    v_number NUMBER := 635;
    v_temp NUMBER;
    v_digit NUMBER;
    v_sum NUMBER := 0;
BEGIN
    v_temp := v_number;

    WHILE v_temp > 0 LOOP
        v_digit := MOD(v_temp, 10);
        v_sum := v_sum + v_digit;
        v_temp := TRUNC(v_temp / 10);
    END LOOP;

    DBMS_OUTPUT.PUT_LINE('Number: ' || v_number);
    DBMS_OUTPUT.PUT_LINE('Sum of Digits: ' || v_sum);
END;
/
```

```
Statement processed.
Number: 635
Sum of Digits: 14
```