WEEK-02

### Question 1

Write a Java program to input a number from user and print it into words using for loop. How to display number in words using loop in Java programming.

Logic to print number in words in Java programming.

**Example**

**Input**

1234  
**Output**  
One Two Three Four

Input:

16

Output:

one six

**For example:**

| **Test** | **Input** | **Result** |
| --- | --- | --- |
| 1 | 45 | Four Five |
| 2 | 13 | One Three |
| 3 | 87 | Eight Seven |

Program:

import java.util.Scanner;

public class prog{

public static void main(String[] args){

Scanner input=new Scanner(System.in);

String name="";

String value=input.nextLine();

for(int i=0;i<value.length();i++){

switch((value.charAt(i))){

case '1':

name="One";

break;

case '2':

name="Two";

break;

case '3':

name= "Three";

break;

case '4':

name="Four";

break;

case '5':

name="Five";

break;

case '6':

name="Six";

break;

case '7':

name="Seven";

break;

case '8':

name="Eight";

break;

case '9':

name="Nine";

break;

case '0':

name="Zero";

break;

}

System.out.print(name+" ");

}

}

}

| **Test** | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 1 | 45 | Four Five | Four Five |  | |
|  | 2 | 13 | One Three | One Three |  | |
|  | 3 | 87 | Eight Seven | Eight Seven |  |

### Question 2

You have recently seen a motivational sports movie and want to start exercising regularly. Your coach tells you that it is important to get up early in the morning to exercise. She sets up a schedule for you:

On weekdays (Monday - Friday), you have to get up at 5:00. On weekends (Saturday & Sunday), you can wake up at 6:00. However, if you are on vacation, then you can get up at 7:00 on weekdays and 9:00 on weekends.

Write a program to print the time you should get up.

Input Format

Input containing an integer and a boolean value.

The integer tells you the day it is (1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, 7-Saturday). The boolean is true if you are on vacation and false if you’re not on vacation.

You have to print the time you should get up.

Example Input:

1 false

Output:

6:00

Example Input:

5 false

Output:

5:00

Example Input:

1 true

Output:

9:00

**For example:**

| **Input** | **Result** |
| --- | --- |
| 1 false | 6:00 |
| 5 false | 5:00 |
| 1 true | 9:00 |

Program:

import java.util.Scanner;

public class WakeUpTime{

public static String getwakeUpTime(int day, boolean isVacation){

if(isVacation){

if(day==1|| day==7){

return "9:00";

}else{

return "7:00";

}

}else{

if(day==1||day==7){

return "6:00";

}else{

return "5:00";

}

}

}

public static void main(String[] args){

Scanner scanner=new Scanner(System.in);

int day=scanner.nextInt();

boolean isVacation=scanner.nextBoolean();

System.out.println(getwakeUpTime(day, isVacation));

scanner.close();

}

}

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 1 false | 6:00 | 6:00 |  |
|  | 5 false | 5:00 | 5:00 |  |
|  | 1 true | 9:00 | 9:00 |  |

### Question 3

Write a program that takes as parameter an integer n.

You have to print the number of zeros at the end of the factorial of n.

For example, 3! = 6. The number of zeros are 0. 5! = 120. The number of zeros at the end are 1.

Note: n! < 10^5

Example Input:

3

Output:

0

Example Input:

60

Output:

14

Example Input:

100

Output:

24

Example Input:

1024

Output:

253

**For example:**

| **Input** | **Result** |
| --- | --- |
| 3 | 0 |
| 60 | 14 |
| 100 | 24 |
| 1024 | 253 |

### Program:

### import java.util.Scanner;

### public class prog {

### // Function to return trailing

### // 0s in factorial of n

### public static int findTrailingZeros(int n)

### {

### int count=0;

### for (int i = 5; n / i >= 1;i\*=5){

### count += n / i;

### }

### return count;

### }

### // Driver Code

### public static void main(String[] args)

### {

### Scanner scanner=new Scanner(System.in);

### int n=scanner.nextInt();

### System.out.println(findTrailingZeros(n));

### scanner.close();

### }

### }

| **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- |
|  | 3 | 0 | 0 |  |
|  | 60 | 14 | 14 |  |
|  | 100 | 24 | 24 |  |
|  | 1024 | 253 | 253 |  |