Name of the Course : Java for beginners: Step–by–step hands-on guide to Java

Level : Moderate

Tool Stack : Java String and Java Array

Problem Statement : Numerologist want to find the numerological value for a given name.

Note: Store the numerological number and the corresponding character in a 2-D array(2\*26).

Always the given name should be in capital case ,else the name is not valid. Check for the valid name,if the name is invalid print the message "Invalid name".

There should not be any space in the name provided.

For example:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

1 2 3 4 5 8 3 5 1 1 2 3 4 5 7 8 1 2 3 4 6 6 6 5 1 7

Description : Create a class Numerology.java with the main method.

Note : Input should be the any name in capital letter and the output is the numerological number which is sum of digits corresponding to characters in name string.

**Code:**

**import** java.util.Scanner;

**class** Numerology

{

**static** String[][] *arr* = { { "A", "B" ,"C", "D" ,"E", "F" ,"G", "H" ,"I", "J" ,"K", "L" ,"M", "N" ,"O", "P" ,"Q", "R" ,"S", "T" ,"U", "V" ,"W", "X" ,"Y", "Z" },

{ "1", "2" ,"3", "4" ,"5", "8" ,"3", "5" ,"1", "1" ,"2", "3" ,"4", "5" ,"7", "8" ,"1", "2" ,"3", "4" ,"6", "6" ,"6", "5" ,"1", "7" } };

**static** **int** findValue(String n) {

**int** sum=0;

**for**(**int** i=0;i<n.length();i++)

{

**char** c=n.charAt(i);

**if**(!(c>='A' && c<='Z'))

{

**return** -1;

}

}

**for**(**int** i=0;i<n.length();i++)

{

String s=Character.*toString*(n.charAt(i));

**for**(**int** j=0;j<26;j++)

{

**if**(s.equals(*arr*[0][j]))

{

**int** m=Integer.*parseInt*(*arr*[1][j]);

sum=sum+m;

}

}

}

**return** sum;

}

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

{

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("Enter your name:");

String n=sc.next();

**int** t=*findValue*(n);

**if**(t==-1) {

System.***out***.println("Invalid name");

}

**else** {

System.***out***.println("Your numerology no is:"+t);

}

}

}

Pom.xml

<project xmlns=*"http://maven.apache.org/POM/4.0.0"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd"*>

<modelVersion>4.0.0</modelVersion>

<groupId>Numerology</groupId>

<artifactId>Numerology</artifactId>

<version>0.0.1-SNAPSHOT</version>

<build>

<sourceDirectory>src</sourceDirectory>

<plugins>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.0</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

</plugins>

</build>

</project>

Junit Testing

import static org.junit.jupiter.api.Assertions.\*;

import org.junit.jupiter.api.Test;

class NumerologyTest {

@Test

void testFindValue() {

assertEquals(17, Numerology.findValue("LAKHAN"));

assertEquals(-1, Numerology.findValue("Arti"));

}

}

Test Data1

Enter your name:

LAKHAN

Your numerology no is:17

Test Data2

Enter your name:

Arti

Invalid name

Test Data3

Enter your name:

Ant123

Invalid name

Learning outcome: Participant could able to learn how to use 2D Array and String concept.