**JAVA ASSIGNMENT**

1). Java Program to find the grade of a student using switch case. Here we will write a Java program find out students grades using switch case statement. The below table shows the grading system.

package com.torryharris.mainpack;  
  
import java.util.Scanner;  
  
public class Main  
{  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.print("Enter the marks : ");  
 int n=sc.nextInt();  
 switch(n/10)  
 {  
 case 9:  
 System.*out*.println("A");  
 break;  
 case 8:  
 System.*out*.println("B");  
 break;  
 case 7:  
 System.*out*.println("C");  
 break;  
 case 6:  
 System.*out*.println("D");  
 break;  
 case 5:  
 System.*out*.println("E");  
 break;  
 case 4:  
 System.*out*.println("F");  
 break;  
 default:  
 System.*out*.println("Enter a valid marks");  
 }

**OUTPUT:**

Enter the marks : 76

C

(2). Pascal Triangle in Java | Pascal triangle is a triangular array of binomial coefficients. In pascal’s triangle, each number is the sum of the two numbers directly above it.

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

Soln:

package com.torryharris.mainpack;  
public class Main  
{  
 public static void main(String[] args) {

int n = 5;  
 for (int i = 0; i < n; i++) {  
 int num = 1;  
 System.*out*.printf("%" + (n - i) \* 2 + "s", " ");  
 for (int j = 0; j <= i; j++) {  
 System.*out*.printf("%4d", num);  
 num = num \* (i - j) / (j + 1);  
 }  
 System.*out*.println();  
 }  
 }  
}

**OUTPUT:**

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

(3). **A number is said to be a magic number if the sum of its digits is calculated till a single digit recursively by adding the sum of the digits after every addition. If the single-digit comes out to be 1, then the number is a magic number.** Example of magic numbers are:- 10, 19, 28, 55, 1234 e.t.c.

Soln:

package com.torryharris.mainpack;  
public class Main  
{  
 public static void main(String[] args) {

System.out.print("Enter a number : ");  
 Scanner sc = new Scanner(System.in);  
 int n = sc.nextInt();  
 if (isMagic(n))  
 System.out.println(n+" Magic Number");  
  
 else  
 System.out.println(n+" Not a magic Number");  
}  
public static boolean isMagic(int n)  
{  
 int sum = 0;  
 while (n > 0 || sum > 9)  
 {  
 if (n == 0)  
 {  
 n = sum;  
 sum = 0;  
 }  
 sum += n % 10;  
 n /= 10;  
 }  
 return (sum == 1);

}

}

**OUTPUT:**

Enter a number : 92

92 Not a magic Number

Enter a number : 10

10 Magic Number

(4). The highest common factor (HCF) of two or more numbers is the greatest number which divides each of them exactly. Greatest Common Measure(GCM) and Greatest Common Divisor(GCD) are the other terms used to refer to HCF.

Soln:

package com.torryharris.mainpack;  
public class Main  
{  
 public static void main(String[] args) {

System.*out*.print("Enter First Number: ");  
 num1 = scanner.nextInt();  
 System.*out*.print("Enter Second Number: ");  
 num2 = scanner.nextInt();  
 scanner.close();  
  
 temp1 = num1;  
 temp2 = num2;  
  
 while(temp2 != 0){  
 temp = temp2;  
 temp2 = temp1%temp2;  
 temp1 = temp;  
 }  
  
 hcf = temp1;  
 lcm = (num1\*num2)/hcf;  
  
 System.*out*.println("HCF of input numbers: "+hcf);  
 System.*out*.println("LCM of input numbers: "+lcm);  
 }  
}

**OUTPUT:**

Enter First Number: 10

Enter Second Number: 35

HCF of input numbers: 5

LCM of input numbers: 70

5.palindrone:

package com.company;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

// write your code here

Scanner sc=new Scanner(System.in);

int m=sc.nextInt();

int rem=0,pal=0,temp;

temp=m;

while(m!=0)

{

rem=m%10;

pal=(pal\*10)+rem;

m=m/10;

}

if(temp==pal)

{

System.out.println(temp+" is a palindrome. its reverse is "+pal);

}

else

System.out.println(temp+" is not a palindrome. its reverse is "+pal);

}

}

Output:

345

345 is not a palindrome. its reverse is 543

Process finished with exit code 0

121

121 is a palindrome. its reverse is 121

Process finished with exit code 0

6. Krishnamurthy number

package com.company;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

// write your code here

Scanner sc=new Scanner(System.in);

int num=sc.nextInt();

int rem,sum=0;

int temp=num;

while(num!=0)

{

rem=num%10;

sum=sum+fact(rem);

num=num/10;

}

if(sum==temp)

System.out.println(temp+" is a krishnamurthy number");

else

System.out.println(temp+" is not a krishnamurthy number");

}

public static int fact(int fac)

{

int r=1;

for(int i=1;i<=fac;i++)

{

r=r\*i;

}

return r;

}

}

output:

145

145 is a krishnamurthy number

Process finished with exit code 0

7. sum of even digit in a number

package com.company;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

// write your code here

Scanner sc=new Scanner(System.in);

int num=sc.nextInt();

int rem,sum=0;

int temp=num;

while(num!=0)

{

rem=num%10;

if(rem%2==0)

sum+=rem;

num=num/10;

}

System.out.println("sum of even digit in "+temp+" is : "+sum);

}

}

output:

7854563

sum of even digit in 7854563 is : 18

Process finished with exit code 0

8.check a number is Disarium or not:

package com.company;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

// write your code here

Scanner sc=new Scanner(System.in);

int num=sc.nextInt();

int rem,sum=0,digit=0;

int temp=num;

int num1=num;

while(num>0)

{

num=num/10;

digit++;

}

// System.out.println(digit);

while(num1>0)

{

rem=num1%10;

sum=sum+ (int)Math.pow(rem,digit);

digit--;

num1=num1/10;

}

System.out.println(sum);

System.out.println(temp);

if (sum==temp)

System.out.println(temp+" is a Disarium number");

else

System.out.println(temp+" is not a disarium number");

}

}

output:

175

175

175

175 is a Disarium number

Process finished with exit code 0

9.

package com.thbs;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter number to find the series: ");

int n = sc.nextInt();

int num0 = 0, num1 = 1;

System.out.print(num0 + " " + num1);

for (int i = 2; i < n; i++) {

int num2 = num0 + num1;

num0 = num1;

num1 = num2;

System.out.print(" " + num2);

}

}

}

Output:

Enter number to find the series:

12

0 1 1 2 3 5 8 13 21 34 55 89

10.

package com.thbs;

public class Main {

public static void main(String[] args) {

System.out.print("Enter Celsius : ");

float c = sc.nextFloat();

float f = ((c \* 9) / 5 + 32);

System.out.println("Value of Celsius in Fahrenheit : " + f);

}

}

Output:

Enter Celsius : 33.5

Value of Celsius in Fahrenheit : 92.3

11.

//------largest number in an array------

System.out.print("enter size of array :");

int n = sc.nextInt();

int arr[] = new int[n];

System.out.println("enter elements:");

for(int i=0;i<n;i++){

arr[i]=sc.nextInt();

}

int lar =0;

for(int i=0;i<n;i++) {

if (arr[i] > lar) {

lar = arr[i];

}

}

System.out.println("Largest number in array is :"+lar);

Output:

enter size of array :4

enter elements:

12

76

34

98

Largest number in array is :98

12.

package com.thbs;

import java.lang.\*;

public class Main {

public static void mergeArrays(int[] arr1, int[] arr2, int[] arr3)

{

int i = 0, j = 0, k = 0;

while (i<arr1.length && j <arr2.length)

{

//checking whe

if (arr1[i] < arr2[j])

arr3[k++] = arr1[i++];

else

arr3[k++] = arr2[j++];

}

// Store remaining elements of first array

while (i < arr1.length)

arr3[k++] = arr1[i++];

// Store remaining elements of second array

while (j < arr2.length)

arr3[k++] = arr2[j++];

}

public static void main (String[] args)

{

int[] arr1 = {-1, 3, 15, 87};

int[] arr2 = {-2, -4, 16, 68};

int[] arr3 = new int[arr1.length+arr2.length];

int n =arr1.length+arr2.length;

mergeArrays(arr1, arr2, arr3);

System.out.println("Array after merging");

for (int i=0; i < n; i++)

System.out.print(arr3[i] + " ");

}

}

Output:

Array after merging

-2 -4 -1 3 15 16 68 87

package com.torryharris;

public class Main {

public static void main(String[] args) {

int[] arr={10,10,9,8,20,30,40,9,8};

Prog13 ob1 = new Prog13();

System.out.println("=======================================================================\n"+"OUTPUT 13");

ob1.duplicate(arr);

Prog14 ob2 = new Prog14();

System.out.println("=======================================================================\n"+"OUTPUT 14");

ob2.rmv\_duplicate(arr);

Prog15 ob3=new Prog15();

Integer[] list=new Integer[arr.length];

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

list[i]=Integer.valueOf(arr[i]);

}

System.out.println("=======================================================================\n"+"OUTPUT 15");

ob3.add\_ele(list);

Prog16 ob4=new Prog16();

System.out.println("=======================================================================\n"+"OUTPUT 16");

for(int i=1;i<=1000;i++){

ob4.arm\_strong(i);

}

}

}

----------------------------------------------------------------------------------------------------------------------------------------

package com.torryharris;

import java.util.\*;

public class Prog13 {

public void duplicate(int[] arr){

Arrays.sort(arr);

int prev=arr[0]-1;

int count=0;

System.out.println("The duplicate elements are : ");

for (int j : arr) {

if (j == prev) {

System.out.print(j + "\t");

++count;

} else

prev = j;

}

System.out.println();

System.out.println(count);

}

}

-----------------------------------------------------------------------------------------------------

package com.torryharris;

import java.util.Arrays;

import java.util.LinkedHashSet;

public class Prog14 {

public void rmv\_duplicate(int[] arr){

Arrays.sort(arr);

LinkedHashSet<Integer> set=new LinkedHashSet<>();

for (int i:arr){

set.add(i);

}

System.out.println(set);

}

}

--------------------------------------------------------------------------------------------------

package com.torryharris;

import java.util.\*;

public class Prog15 {

public void add\_ele(Integer[] arr){

List<Integer> list=new ArrayList<>(Arrays.asList(arr));

Scanner sc=new Scanner(System.in);

System.out.println("Enter the position of the array : ");

int pos=sc.nextInt();

System.out.println("Enter the value : ");

int ele= sc.nextInt();

list.add(pos-1,ele);

arr=list.toArray(arr);

System.out.println(Arrays.toString(arr));

sc.close();

}

}

----------------------------------------------------------------------------------------------------

package com.torryharris;

import java.util.\*;

public class Prog16 {

public void arm\_strong(int num){

char[] ch= (""+num).toCharArray();

int len=ch.length;

int sum=0;

for (char c:ch){

int n=Integer.parseInt(String.valueOf(c));

sum += Math.pow(n,len);

}

if (sum==num)

System.out.println(num+" is an armstrong number");

}

}

-------------------------------------------------------------------------------------------------

=======================================================================

OUTPUT 13

The duplicate elements are :

8 9 10

3

=======================================================================

OUTPUT 14

[8, 9, 10, 20, 30, 40]

=======================================================================

OUTPUT 15

Enter the position of the array :

2

Enter the value :

50

[8, 50, 8, 9, 9, 10, 10, 20, 30, 40]

=======================================================================

OUTPUT 16

1 is an armstrong number

2 is an armstrong number

3 is an armstrong number

4 is an armstrong number

5 is an armstrong number

6 is an armstrong number

7 is an armstrong number

8 is an armstrong number

9 is an armstrong number

153 is an armstrong number

370 is an armstrong number

371 is an armstrong number

407 is an armstrong number

17.In mathematics, a number is called an Automorphic number if the square of the number ends with the same number. Example of Automorphic numbers are:- 5, 6, 25, 76, e.t.c..

Ans:

package com.TH;  
import java.util.Scanner;  
  
public class Main {  
  
 public static void main(String args[]){  
  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Enter a number");  
 int num = in.nextInt();  
  
 int c=0, sqr = num\*num;  
 int temp =num;   
  
   
 while(temp>0){  
 c++;  
 temp=temp/10;  
 }  
  
 int lastSquareDigits = (int) (sqr%(Math.*pow*(10,c)));  
  
 if(num == lastSquareDigits)  
 System.*out*.println("Automorphic number");  
 else  
 System.*out*.println("Not an Automorphic number");  
 }  
}

output:

Enter a number

25

Automorphic number

18.

we will develop a Java program for a BMI calculator.

The BMI stands for Body Mass Index. It is a measure of health based on height and weight. It can be calculated by taking the weight in kilograms and dividing it by the square of your height in meters.

Formula for Calculating BMI in Metric Units,

BMI = (Weight in Kg) / (Height in Meters \* Height in Meters)

For example,

weight = 75 kg, height = 1.5 m

BMI = 75 / (1.5\*1.5) = 33.33

Using the range of BMI, individuals are classified as underweight, normal or overweight. Its value is in a specific range for a healthy individual. The following table shows the main BMI categories,

BMI Range Category

< 18.5 Thinness

18.5 – 25 Normal

25 – 30 Overweight

> 30 Obese

BMI calculator finds whether the person is underweight, normal, overweight, or obese. The person who wants to lose their want can use a weight loss calculator which can tell the daily caloric requirements, and many easy tips to lose weight.

Ans:

package com.TH;  
import java.util.Scanner;  
public class Main {  
 public static void main(String args[]) {  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.print("Input weight in kilogram: ");  
 double weight = sc.nextDouble();  
 System.*out*.print("\nInput height in meters: ");  
 double height = sc.nextDouble();  
 double BMI = weight / (height \* height);  
 if(BMI<18.5)  
 System.*out*.println("thickness");  
 else if (BMI>18.5&&BMI<25)  
 System.*out*.println("normal");  
 else if (BMI>25&&BMI<30)  
 System.*out*.println("over weight");  
 else  
 System.*out*.println("obese");  
 }  
}

output:

Input weight in kilogram:

75

Input height in meters:

1.65

over weight

Input weight in kilogram:

63

Input height in meters:

1.63

normal

19.Perfect number:- A number whose factors sum except itself, is equal to the same number is called a perfect number.

Ans:

package com.TH;  
import java.util.Scanner;  
  
public class Main {  
 private static Scanner *sc*;  
  
 public static void main(String[] args) {  
 int i, Number, Sum = 0 ;  
 *sc* = new Scanner(System.*in*);  
 System.*out*.println("\n Please Enter any Number: ");  
 Number = *sc*.nextInt();  
  
 for(i = 1 ; i < Number ; i++) {  
 if(Number % i == 0) {  
 Sum = Sum + i;  
 }  
 }  
 if (Sum == Number) {  
 System.*out*.format("\n% d is a Perfect Number", Number);  
 }  
 else {  
 System.*out*.format("\n% d is NOT a Perfect Number", Number);  
 }  
 }  
}

output:

Please Enter any Number:

6

6 is a Perfect Number

20.Write a Java program to find the second largest number in Java.

Ans:

package com.TH;  
  
import java.util.Scanner;  
  
public class Main {  
  
 public static void main(String[] args) {  
 System.*out*.println("enter no.of.element from array");  
 Scanner s=new Scanner(System.*in*);  
 int n=s.nextInt();  
  
 int []arr =new int[n] ;  
 System.*out*.println("enter elements:");  
 for (int i=0;i<n;i++)  
 {  
 arr[i]=s.nextInt();  
 }  
  
 int largest = arr[0];  
 int secondLargest = arr[0];  
  
 System.*out*.println("The given array is:" );  
 for (int i = 0; i < arr.length; i++) {  
 System.*out*.print(arr[i]+"\t");  
 }  
 for (int i = 0; i < arr.length; i++) {  
  
 if (arr[i] > largest) {  
 secondLargest = largest;  
 largest = arr[i];  
  
 } else if (arr[i] > secondLargest) {  
 secondLargest = arr[i];  
  
 }  
 }  
  
 System.*out*.println("\nSecond largest number is:" + secondLargest);  
  
 }  
}

output:

enter no.of.element from array

5

enter elements:

65

87

67

54

35

The given array is:

65 87 67 54 35

Second largest number is:67