JAVA Programming Code:

Q. How to convert Fahrenheit to Celsius Program

Code:

import java.util.\*;

class FahrenheitToCelsius {

public static void main(String[] args) {

float temperatue;

Scanner in = new Scanner(System.in);

System.out.println("Enter temperatue in Fahrenheit");

temperatue = in.nextInt();

temperatue = ((temperatue - 32)\*5)/9;

System.out.println("Temperatue in Celsius = " + temperatue);

}

}

Q. Find Largest no in java

Code:

import java.util.Scanner;

class LargestOfThreeNumbers

{

public static void main(String args[])

{

int x, y, z;

System.out.println("Enter three integers ");

Scanner in = new Scanner(System.in);

x = in.nextInt();

y = in.nextInt();

z = in.nextInt();

if ( x > y && x > z )

System.out.println("First number is largest.");

else if ( y > x && y > z )

System.out.println("Second number is largest.");

else if ( z > x && z > y )

System.out.println("Third number is largest.");

else

System.out.println("Entered numbers are not distinct.");

}

Q. Find factorial for given no Program

Code:

import java.util.Scanner;

class Factorial

{

public static void main(String args[])

{

int n, c, fact = 1;

System.out.println("Enter an integer to calculate it's factorial");

Scanner in = new Scanner(System.in);

n = in.nextInt();

if ( n < 0 )

System.out.println("Number should be non-negative.");

else

{

for ( c = 1 ; c <= n ; c++ )

fact = fact\*c;

System.out.println("Factorial of "+n+" is = "+fact);

}

}

}

Q. Print Reverse number

Code:

import java.util.Scanner;

class ReverseNumber

{

public static void main(String args[])

{

int n, reverse = 0;

System.out.println("Enter the number to reverse");

Scanner in = new Scanner(System.in);

n = in.nextInt();

while( n != 0 )

{

reverse = reverse \* 10;

reverse = reverse + n%10;

n = n/10;

}

System.out.println("Reverse of entered number is "+reverse);

}

}

Q. Check no is Armstrong or not

Code:

import java.util.Scanner;

class ArmstrongNumber

{

public static void main(String args[])

{

int n, sum = 0, temp, remainder, digits = 0;

Scanner in = new Scanner(System.in);

System.out.println("Input a number to check if it is an Armstrong

number");

n = in.nextInt();

temp = n;

// Count number of digits

while (temp != 0) {

digits++;

temp = temp/10;

}

temp = n;

while (temp != 0) {

remainder = temp%10;

sum = sum + power(remainder, digits);

temp = temp/10;

}

if (n == sum)

System.out.println(n + " is an Armstrong number.");

else

System.out.println(n + " is not an Armstrong number.");

}

static int power(int n, int r) {

int c, p = 1;

for (c = 1; c <= r; c++)

p = p\*n;

return p;

} }

Q. Print reverse string

Code:

import java.util.\*;

class ReverseString

{

public static void main(String args[])

{

String original, reverse = "";

Scanner in = new Scanner(System.in);

System.out.println("Enter a string to reverse");

original = in.nextLine();

int length = original.length();

for ( int i = length - 1 ; i >= 0 ; i-- )

reverse = reverse + original.charAt(i);

System.out.println("Reverse of entered string is: "+reverse);

}

}

//Using Internal java Methog

class InvertString

{

public static void main(String args[])

{

StringBuffer a = new StringBuffer("Java programming is fun");

System.out.println(a.reverse());

}

}

Q. Check Given No is palindrome or Not

Code:

import java.util.\*;

class Palindrome

{

public static void main(String args[])

{

String original, reverse = "";

Scanner in = new Scanner(System.in);

System.out.println("Enter a string to check if it is a palindrome");

original = in.nextLine();

int length = original.length();

for ( int i = length - 1; i >= 0; i-- )

reverse = reverse + original.charAt(i);

if (original.equals(reverse))

System.out.println("Entered string is a palindrome.");

else

System.out.println("Entered string is not a palindrome.");

}

}

Q. How to multiply two matrix

Code:

import java.util.Scanner;

class MatrixMultiplication

{

public static void main(String args[])

{

int m, n, p, q, sum = 0, c, d, k;

Scanner in = new Scanner(System.in);

System.out.println("Enter the number of rows and columns of first

matrix");

m = in.nextInt();

n = in.nextInt();

int first[][] = new int[m][n];

System.out.println("Enter the elements of first matrix");

for ( c = 0 ; c < m ; c++ )

for ( d = 0 ; d < n ; d++ )

first[c][d] = in.nextInt();

System.out.println("Enter the number of rows and columns of second

matrix");

p = in.nextInt();

q = in.nextInt();

if ( n != p )

System.out.println("Matrices with entered orders can't be multiplied

with each other.");

else

{

int second[][] = new int[p][q];

int multiply[][] = new int[m][q];

System.out.println("Enter the elements of second matrix");

for ( c = 0 ; c < p ; c++ )

for ( d = 0 ; d < q ; d++ )

second[c][d] = in.nextInt();

for ( c = 0 ; c < m ; c++ )

{

for ( d = 0 ; d < q ; d++ )

{

for ( k = 0 ; k < p ; k++ )

{ sum = sum + first[c][k]\*second[k][d];

}

multiply[c][d] = sum;

sum = 0;

}

}

System.out.println("Product of entered matrices:-");

for ( c = 0 ; c < m ; c++ )

{

for ( d = 0 ; d < q ; d++ )

System.out.print(multiply[c][d]+"\t");

System.out.print("\n");

}

}

}

}

Q, How to create method

Code:

class Methods {

// Constructor method

Methods() {

System.out.println("Constructor method is called when an object of it's

class is created");

}

// Main method where program execution begins

public static void main(String[] args) {

staticMethod();

Methods object = new Methods();

object.nonStaticMethod();

}

// Static method

static void staticMethod() {

System.out.println("Static method can be called without creating

object");

}

// Non static method

void nonStaticMethod() {

System.out.println("Non static method must be called by creating an

object");

}

}

Q. How to create Multiple class

Code:

class Computer {

Computer() {

System.out.println("Constructor of Computer class.");

}

void computer\_method() {

System.out.println("Power gone! Shut down your PC soon...");

}

public static void main(String[] args) {

Computer my = new Computer();

Laptop your = new Laptop();

my.computer\_method();

your.laptop\_method();

}

}

class Laptop {

Laptop() {

System.out.println("Constructor of Laptop class.");

}

void laptop\_method() {

System.out.println("99% Battery available.");

}

}

Q. How to create constructor overloading

Code:

class Language {

String name;

Language() {

System.out.println("Constructor method called.");

}

Language(String t) {

name = t;

}

public static void main(String[] args) {

Language cpp = new Language();

Language java = new Language("Java");

cpp.setName("C++");

java.getName();

cpp.getName();

}

void setName(String t) {

name = t;

}

void getName() {

System.out.println("Language name: " + name);

}

}

Q. How to create Interface

Code:

interface Info {

static final String language = "Java";

public void display();

}

class Simple implements Info {

public static void main(String []args) {

Simple obj = new Simple();

obj.display();

}

// Defining method declared in interface

public void display() {

System.out.println(language + " is awesome");

}

Q. How to Generate random number

Code:

import java.util.\*;

class RandomNumbers {

public static void main(String[] args) {

int c;

Random t = new Random();

// random integers in [0, 100]

for (c = 1; c <= 10; c++) {

System.out.println(t.nextInt(100));

}

}

}

Q. Leaner search Program

Code:

import java.util.Scanner;

class LinearSearch

{

public static void main(String args[])

{

int c, n, search, array[];

Scanner in = new Scanner(System.in);

System.out.println("Enter number of elements");

n = in.nextInt();

array = new int[n];

System.out.println("Enter " + n + " integers");

for (c = 0; c < n; c++)

array[c] = in.nextInt();

System.out.println("Enter value to find");

search = in.nextInt();

for (c = 0; c < n; c++)

{

if (array[c] == search) /\* Searching element is present \*/

{

System.out.println(search + " is present at location " + (c + 1) +

".");

break;

}

}

if (c == n) /\* Searching element is absent \*/

System.out.println(search + " is not present in array.");

}

}

Q. Binary search

Code:

import java.util.Scanner;

class BinarySearch

{

public static void main(String args[])

{

int c, first, last, middle, n, search, array[];

Scanner in = new Scanner(System.in);

System.out.println("Enter number of elements");

n = in.nextInt();

array = new int[n];

System.out.println("Enter " + n + " integers");

for (c = 0; c < n; c++)

array[c] = in.nextInt();

System.out.println("Enter value to find");

search = in.nextInt();

first = 0;

last = n - 1;

middle = (first + last)/2;

while( first <= last )

{

if ( array[middle] < search )

first = middle + 1;

else if ( array[middle] == search )

{

System.out.println(search + " found at location " + (middle + 1) +

".");

break;

}

else

last = middle - 1;

middle = (first + last)/2;

}

if ( first > last )

System.out.println(search + " is not present in the list.\n");

}

}

Q. Bubble sort Program in java

Code:

import java.util.Scanner;

class BubbleSort {

public static void main(String []args) {

int n, c, d, swap;

Scanner in = new Scanner(System.in);

System.out.println("Input number of integers to sort");

n = in.nextInt();

int array[] = new int[n];

System.out.println("Enter " + n + " integers");

for (c = 0; c < n; c++)

array[c] = in.nextInt();

for (c = 0; c < ( n - 1 ); c++) {

for (d = 0; d < n - c - 1; d++) {

if (array[d] > array[d+1]) /\* For descending order use < \*/

{

swap = array[d];

array[d] = array[d+1];

array[d+1] = swap;

}

}

}

System.out.println("Sorted list of numbers");

for (c = 0; c < n; c++)

System.out.println(array[c]);

}

}

Q. How to check Regular expression in java

Program

import java.util.regex.\*;

public class RegexExample1{

public static void main(String args[]){

//1st way

Pattern p = Pattern.compile(".s");//. represents single character

Matcher m = p.matcher("as");

boolean b = m.matches();

//2nd way

boolean b2=Pattern.compile(".s").matcher("as").matches();

//3rd way

boolean b3 = Pattern.matches(".s", "as");

System.out.println(b+" "+b2+" "+b3);

}}

Q. How to join thread in java program

class TestJoinMethod1 extends Thread{

public void run(){

for(int i=1;i<=5;i++){

try{

Thread.sleep(500);

}catch(Exception e){System.out.println(e);}

System.out.println(i);

}

}

public static void main(String args[]){

TestJoinMethod1 t1=new TestJoinMethod1();

TestJoinMethod1 t2=new TestJoinMethod1();

TestJoinMethod1 t3=new TestJoinMethod1();

t1.start();

try{

t1.join();

}catch(Exception e){System.out.println(e);}

t2.start();

t3.start();

}

}