Run and verified these program in java, as its taking many repositories hence updating all in this doc. Please email me if you want Java files for it

1.

public class acad {

public void sumTotal() {

int a1 = 10;

int a2 = 15;

int a3 = a1 + a2;

System.out.println("Total : " + a3);

}

public static void main(String args[]) {

acad t1 = new acad();

t1.sumTotal();

}

}

2.

import java.util.Scanner;

public class acad {

public void sumTotal() {

Scanner s = new Scanner(System.in);

System.out.print("Enter first number: ");

int a1 = s.nextInt();

System.out.print("Enter second number: ");

int a2 = s.nextInt();

int a3 = a1 + a2;

System.out.println("Total : " + a3);

}

public static void main(String[] args) {

acad t1 = new acad();

t1.sumTotal();

}

}

3.

import java.util.Scanner;

public class Addition {

public void sum() {

Scanner s = new Scanner(System.in);

System.out.print("Enter first number: ");

int a1 = s.nextInt();

System.out.print("Enter second number: ");

int a2 = s.nextInt();

int a3 = a1 + a2;

System.out.println("First Number is : " + a1);

System.out.println("Second Number is : " + a2);

System.out.println("Sum is : " + a3);

}

public static void main(String[] args) {

Addition t1 = new Addition();

t1.sum();

}

}

4.

import java.util.Scanner;

public class EvenOdd {

public void sum() {

Scanner s = new Scanner(System.in);

System.out.print("Enter a number: ");

int num1 = s.nextInt();

if (num1%2 == 0)

{

System.out.println(num1 + " is Even");

}

else

{

System.out.println(num1 + " is odd");

}

}

public static void main(String[] args) {

EvenOdd t1 = new EvenOdd();

t1.sum();

}

}

5.

import java.util.Scanner;

public class Multiple {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

System.out.print("Enter a number: ");

int num1 = s.nextInt();

for (int i = 1;i<=10;i++)

{

System.out.println(+num1+ " x "+i+ " " + i\*num1);

}

}

}

6.

class TestOverLoad {

static int sum(int a,int b)

{

return a+b;

}

static int sum(int a,int b,int c)

{

return a+b+c;

}

}

class TestOverloading1 {

public static void main(String[] args)

{

System.out.println(TestOverLoad.sum(11,11));

System.out.println(TestOverLoad.sum(11,11,11));

}

}

7.

In a class, there can be several methods sharing the same name but differ in

Parameter types

Number of parameters

Order of the parameters declared in the method

By depending on the parameters provided for the method, in the run time, compiler determines which version of the method to execute.

An overloaded method may or may not have different return types. But return type alone is not sufficient for the compiler to determine which method is to be executed at run time.

8.

public class BubbleSort {

public static void main(String []args) {

int array[] = new int[]{7,5,3,1,4,8,9,2,6};

int len = array.length;

int temp = 0;

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

for(int j=1; j < (len-i); j++){

if(array[j-1] < array[j]){

temp = array[j-1];

array[j-1] = array[j];

array[j] = temp;

}

}

}

}

}