CSE-2005

Assignment-1

1. Let us consider two objects Samsung Galaxy S4 and iPhone. Suppose Samsung Galaxy S4 have some properties like width = “6.98 cms”, height = “13.6 cm”, OS = “Android”, brand = “Samsung”, price = “1000$” and actions are call(), sendMessage(), browser(), share(). Now, suppose iPhone has some properties such as width = “5.86 cm”, height = “12.3 cms”, OS = “iOS”, brand = “Apple”, price = “1200$” and actions are call(), sendMessage(), browse(), share(). Both objects have some different properties and actions but the type is the same “Phone”. Create a class named as “Phone”, create objects to represent the scenario and print.

class Phone {

double width;

double height;

String Os ="";

String brand ="";

double price;

void setwidth (double width) {

this.width=width;

}

void setheight (double height) {

this.height=height;

}

void setos (String Os) {

this.Os=Os;

}

void setbrand (String brand) {

this.brand=brand;

}

void setprice(double price) {

this.price=price;

}

double getwidth (){

return width;

}

double getheight(){

return height;

}

String getos() {

return Os;

}

String getbrand() {

return brand;

}

double getprice() {

return price;

}

void call() {

System.out.println("Phone has call option");

System.out.println();

}

void sendmessage() {

System.out.println("Phone has sendmessage option");

System.out.println();

}

void browser() {

System.out.println("Phone has browser option");

System.out.println();

}

void share() {

System.out.println("Phone has share action");

System.out.println();

}

}

public class Phones{

public static void main(String [] args) {

Phone Samsung\_Galaxy\_s4 = new Phone();

Phone iPhone = new Phone();

Samsung\_Galaxy\_s4.setwidth(6.98);

Samsung\_Galaxy\_s4.setheight(13.6);

Samsung\_Galaxy\_s4.setos("Android");

Samsung\_Galaxy\_s4.setbrand("Samsung");

System.out.println("For Samsung\_Galaxy\_s4");

System.out.println();

System.out.println();

double w =Samsung\_Galaxy\_s4.getwidth();

System.out.println("width of Samsung\_Galaxy\_s4 is " + w + " cms");

System.out.println();

double h=Samsung\_Galaxy\_s4.getheight();

System.out.println("height of Samsung\_Galaxy\_s4 is " + h + " cms");

System.out.println();

String os =Samsung\_Galaxy\_s4.getos();

System.out.println("OS of Samsung\_Galaxy\_s4 is "+ os);

System.out.println();

String brand=Samsung\_Galaxy\_s4.getbrand();

System.out.println("Brand of Samsung\_Galaxy\_s4 is "+ brand);

System.out.println();

double price = Samsung\_Galaxy\_s4.getprice();

System.out.println("price of Samsung\_Galaxy\_s4 is "+ price + "$");

System.out.println();

Samsung\_Galaxy\_s4.call();

Samsung\_Galaxy\_s4.sendmessage();

Samsung\_Galaxy\_s4.browser();

Samsung\_Galaxy\_s4.share();

System.out.println("For iPhone");

System.out.println();

System.out.println();

iPhone.setwidth(5.86);

iPhone.setheight(12.3);

iPhone.setos("iOS");

iPhone.setbrand("Apple");

iPhone.setprice(1200);

w = iPhone.getwidth();

System.out.println("width of iPhone is " + w + " cms");

System.out.println();

h = iPhone.getheight();

System.out.println("height of iPhone is " + h + " cms");

System.out.println();

os = iPhone.getos();

System.out.println("OS of iPhone is "+ os);

System.out.println();

brand = iPhone.getbrand();

System.out.println("Brand of iPhone is "+ brand);

System.out.println();

price = iPhone.getprice();

System.out.println("price of iPhone is "+ price + "$");

System.out.println();

iPhone.call();

iPhone.sendmessage();

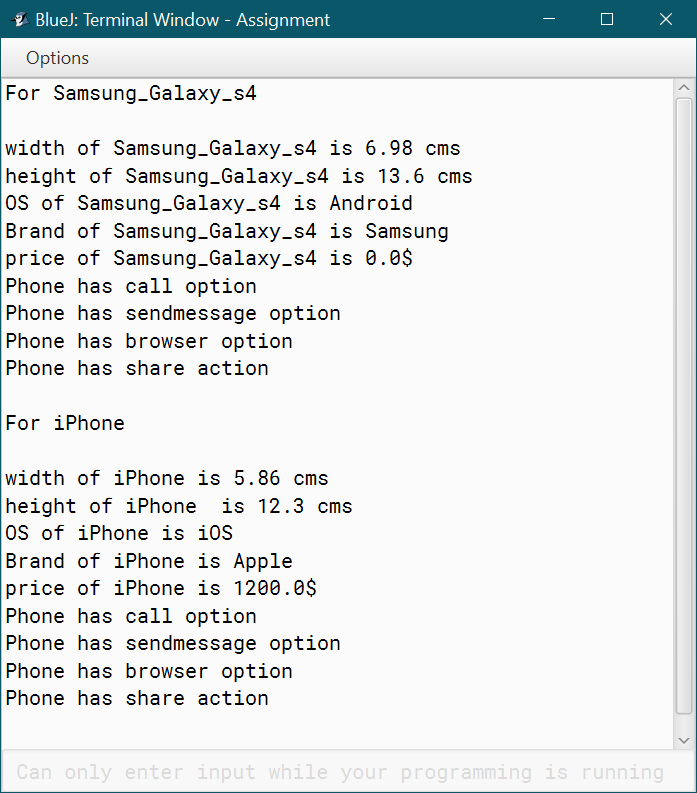
iPhone.browser();

iPhone.share();

}

}

Output :-



1. Consider two different objects one boy and one girl. The boy has some properties like hairColor = “black”, eyeColor = “black”, skinColor = “Fair”, height = “5.10 inch”, weight = “65 kg” and actions are read(), play(), sleep(), walk(). A girl has some properties like hairColor = “brown”, eyeColor = “brown”, skinColor = “milky white”, height = “5.4 inch”, weight = “50 kg” and actions are read(), play(), sleep(), walk(). But the type of both boy and girl is the same. Type is “Person”. Create a class named as “Person”, create objects to represent the scenario and print.

class Person{

String haircolour = "";

String eyecolour = "";

String skincolour = "";

double height;

double weight;

Person(double weight, double height){

this.weight=weight;

this.height=height;

}

void setcolour(String haircolour, String eyecolour, String skincolour){

this.haircolour=haircolour;

this.eyecolour=eyecolour;

this.skincolour=skincolour;

}

void getproperties(){

System.out.println("Hair colour of a person is " + haircolour);

System.out.println("Eye colour of a person is " + eyecolour);

System.out.println("Skin colour of a person is " + skincolour);

System.out.println("Height of a person is " + height + "inch");

System.out.println("Weight of a person is " + weight + "Kg");

}

void read(){

System.out.println("Person can read");

}

void play(){

System.out.println("Person can play");

}

void sleep(){

System.out.println("Person can sleep");

}

void walk(){

System.out.println("Person can walk");

}

void getactions(){

read();

play();

sleep();

walk();

}

}

public class Persons{

public static void main (String[] args) {

Person boy = new Person(5.10, 65);

Person girl = new Person(5.4, 50);

boy.setcolour("black", "black", "Fair");

girl.setcolour("brown", "brown", "milky white");

System.out.println("Properties and actions of boy");

System.out.println();

boy.getproperties();

boy.getactions();

System.out.println();

System.out.println();

System.out.println("Properties and actions of girl");

System.out.println();

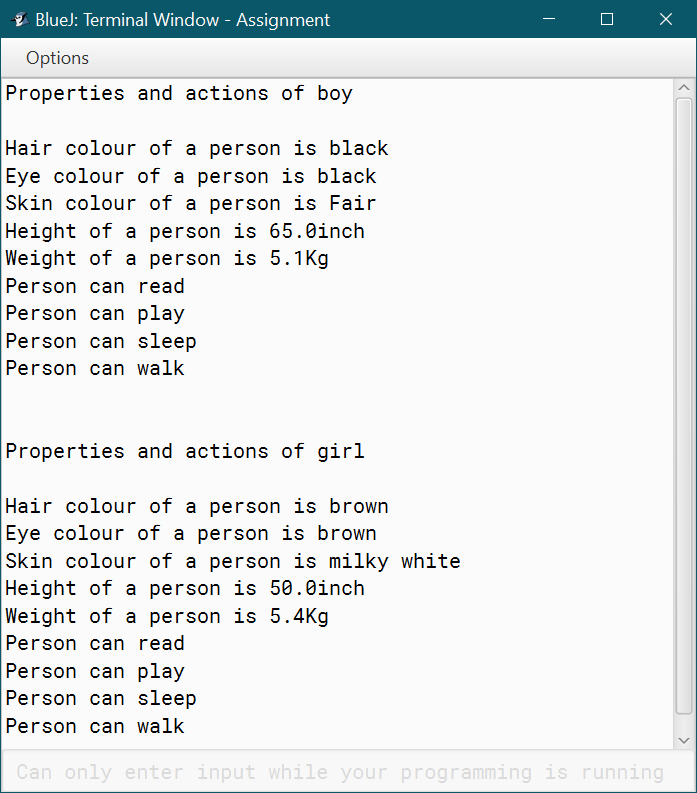
girl.getproperties();

girl.getactions();

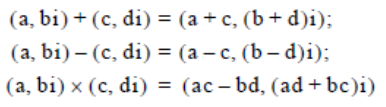
}

}

Output:-



1. Create a class to represent complex numbers. Complex numbers have two parts: a real part and an imaginary part. So a complex number value could be (1, 1.5i ), where 1 is the real part and 1.5i is the imaginary part. You should be able to add, subtract, and multiply two complex numbers. The rules for these operations are:



class Complexnum {

double real, imaginary;

Complexnum(double real, double imaginary) {

this.real = real;

this.imaginary = imaginary;

}

Complexnum(){

}

Complexnum add(Complexnum n1, Complexnum n2) {

Complexnum temp = new Complexnum();

temp.real = n1.real + n2.real;

temp.imaginary = n1.imaginary + n2.imaginary;

return temp;

}

Complexnum sub(Complexnum n1, Complexnum n2){

Complexnum temp = new Complexnum();

temp.real = n1.real - n2.real;

temp.imaginary = n1.imaginary - n2.imaginary;

return temp;

}

Complexnum multiply(Complexnum n1, Complexnum n2){

Complexnum temp = new Complexnum();

temp.real=(n1.real\*n2.real) - (n1.imaginary\*n2.imaginary);

temp.imaginary=(n1.real\*n2.imaginary) + (n1.imaginary\*n2.real);

return temp;

}

}

public class Complexoperation{

public static void main(String[] args){

Complexnum n1 = new Complexnum(1, 1.5);

Complexnum n2 = new Complexnum(2, 3.5);

Complexnum n3 = new Complexnum();

System.out.println("1st complex number is (" + n1.real + "," + n1.imaginary + "i) and 2nd complex number is (" + n2.real + "," + n2.imaginary + "i)");

n3=n3.add(n1, n2);

System.out.println("Sum of complex numbers is (" + n3.real + "," + n3.imaginary + "i)");

n3=n3.sub(n1,n2);

System.out.println("Substract of complex numbers is (" + n3.real + "," + n3.imaginary + "i)");

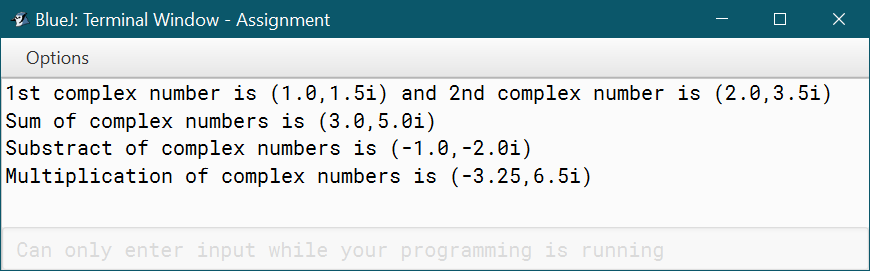
n3=n3.multiply(n1, n2);

System.out.println("Multiplication of complex numbers is (" + n3.real + "," + n3.imaginary + "i)");

}

}

Output:-



* 1. Create a class to represent a pair of integer values. The no-arg constructor should leave both integers at the default value of 0. Include a constructor that can take one value n, setting one of the values of the pair to n and the other to 0. Include a constructor taking two values, setting both values of the pair. Note how useful it is to design a set of overloaded constructors that conveniently initialize the object pair. Have accessor methods that return first and second, the two values of the pair. Also have mutator methods that allow you to change the two values of a pair separately. Write a program to test your new class.
  2. Write a method minMax() that takes as input an array of integers and returns the minimum and the maximum value in the array. The method should return the two integers by using the integer pair class from the previous exercise. Write a program to test your method. ( Use Scanner class to take input or Command Line arguments)

import java.util.Scanner;

class Box{

int a;

int b;

Box(){

this.a = 0;

this.b = 0;

}

Box(int a){

this.a=a;

this.b=0;

}

Box(int a, int b){

this.a=a;

this.b=b;

}

void seta(int a){

this.a=a;

}

void setb(int b){

this.b=b;

}

Box minmax (int [] a){

Box temp = new Box();

temp.seta(a[0]);

for(int i=1;i < a.length;i++){

if(a[i] > temp.a){

temp.seta(a[i]);

}

}

temp.setb(a[0]);

for(int i=1;i<a.length;i++){

if(a[i] < temp.b){

temp.setb(a[i]);

}

}

return temp;

}

}

public class Main{

public static void main (String[] args) {

Box b1 = new Box();

Box b2 = new Box(1);

Box b3 = new Box(3, 4);

Scanner sc = new Scanner(System.in);

System.out.println("Before modificatin");

System.out.println("values of b1 object a = " + b1.a + " b = " + b1.b);

System.out.println("values of b2 object a =" + b2.a + " b = " + b2.b);

System.out.println("values of b3 object a = " + b3.a + " b = " + b3.b);

System.out.println();

System.out.println("After modificatin");

b1.seta(2);

b1.setb(5);

b2.seta(6);

b2.setb(7);

b3.seta(8);

b3.setb(9);

System.out.println("values of b1 object a = " + b1.a + " b = " + b1.b);

System.out.println("values of b2 object a =" + b2.a + " b = " + b2.b);

System.out.println("values of b3 object a = " + b3.a + " b = " + b3.b);

System.out.println();

int c[]=new int[7];

for(int i=0;i<7;i++)

{

c[i]=sc.nextInt();

}

Box q = b1.minmax(c);

System.out.print("Array c = [");

for(int i=0;i<7;i++)

{

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

}

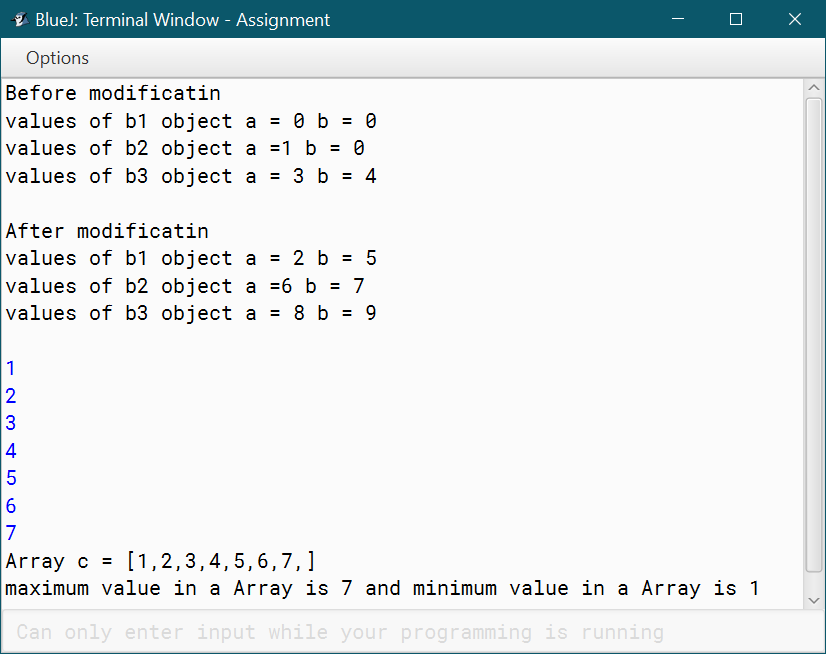
System.out.println("]");

System.out.println("maximum value in a Array is " + q.a + " and minimum value in a Array is " + q.b);

}

}

Output:-



1. Create a class Rectangle that represents a rectangular region of the plane. A rectangle should be described using four integers: two to represent the coordinates of the lower left corner of the rectangle, giving its location; one for the width; and one for the height. Your rectangle class should include

* appropriate constructors,
* a toString() method,
* a method translate() that takes two integer parameters, deltaX and deltaY, used to translate the location of the rectangle,
* a method contains() that takes two integer parameters, xCoord and yCoord, and returns true if the point (xCoord, yCoord) lies within the rectangle, and
* a method intersection() that takes a Rectangle as a parameter and returns a new Rectangle that forms the intersection of the Rectangle upon which the method is operating, and the Rectangle parameter.

Write a program to test the class Rectangle.

class Rectangle{

int x, y, height, width;

Rectangle(int x, int y, int height, int width){

this.x=x;

this.y=y;

this.height=height;

this.width=width;

}

void translate(int x1, int y1){

x=x+x1;

y=y+y1;

}

boolean contains(int x1, int y1){

if(x1<=(x+width) && x1>=x && y1>=y && y1<=(y+height)){

return true;

}

else{

return false;

}

}

public String toString(){

return "Left corner point of Rectangle is (" + x + "," + y + ") and with height " + height + " width " + width;

}

Rectangle intersection(Rectangle r){

int x1 = this.x;

int y2 = this.y;

int x3 = r.x;

int y4 = r.y;

int x2 = x1 + this.width;

int x4 = x3 + r.width;

int y1 = y2 - this.height;

int y3 = y4 - r.width;

int xL = Math.max(x1, x3);

int xR = Math.min(x2, x4);

if (xR <= xL)

return null;

else {

int yT = Math.max(y1, y3);

int yB = Math.min(y2, y4);

if (yB <= yT)

return null;

else

return new Rectangle(xL, yB, xR-xL, yB-yT);

}

}

}

public class Rec{

public static void main (String[] args) {

Rectangle r1 = new Rectangle(5, 4, 9, 8);

Rectangle r2 = new Rectangle(6, 6, 5, 4);

System.out.println("Before translation position of rectangles r1 and r2");

String s = r1.toString();

System.out.println(s);

s = r2.toString();

System.out.println(s);

System.out.println("After translation rectangle r1 with(4,5) and r2 with(1,1) position of rectangel r1 and r2");

r1.translate(4,5);

s = r1.toString();

System.out.println(s);

r2.translate(1,1);

s = r2.toString();

System.out.println(s);

boolean p = r1.contains(19, 9);

System.out.println("Point (19,9) contains in rectangle r1 is " + p);

p = r2.contains(19, 9);

System.out.println("Point (19,9) contains in rectangle r2 is " + p);

//Only for rectangles which are intersected diagolly

Rectangle d =r1.intersection(r2);

s = d.toString();

System.out.println("Intersection rectangle of two rectangles r1 and r2");

System.out.println(s);

}

}

Output:-

