

Programming Assignment 5

Due March 4

1. Implement a class `Point` that represents an integer point in the x-y plane such as (8,12) or (7,23)
Point has two private variables x and y, both of type int.

Also

- have a default constructor that sets x and y both equal to 0 and a
- have a two argument constructor.
- have getters and setters for x and y.
- have a method

`double distance(Point p)`

that returns the distance between two points. So if p is (2,4) and q is (5,6) the p.distance(q) returns the distance between p and q.
(Use the ordinary distance formula.)

- have a method

`Point add(Point p)`

that adds two points so that $(a,b) + (c, d) = (a+c,b+d)$
So if a = (2,3) and b = (4,5) a.add(b) returns (6,8)

- Override

`boolean equals(Object o)`

so that two Points are equal if they have the same x and y coordinates .

- Override

`String toString()`

so that the method returns the point in the form (x,y)
-- parentheses included

Now make a second class Circle that **uses the Point** class.

- a Circle **has-a** radius (int) and a center (a Point)
- The default constructor should make the center the point (0,0) and radius 0
- A two argument constructor should accept a radius and a Point
- Include getters and setters and
- a method
 double area() that returns the area of the circle.
- Override equals from Object so that two Circle objects are equal if they have the same area
- Override toString() so that toString() returns the radius, center, and the area with identifying labels

Make a separate class **TestPointAndCircle** with the following main() method

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

{

    Point a = new Point (3,4);
    Point b = new Point(3,4);
    Point c = new Point();
    Point d = new Point (0,5);
    System.out.println("a = " + a);
    System.out.println("b = " +b);
    System.out.println("c = " + c);
    System.out.println("d = " + d);
    System.out.println("Distance between "+a+" and "+c+" is "+ a.distance(c));
    System.out.println("Sum of "+ a +" and " + d +" is "+ a.add(d));
    System.out.println("Point a equals Point b: "+ a.equals(b));
    System.out.println("Point a equals Point d: "+ a.equals(d));
    System.out.println();
    Circle x = new Circle(1,a);
    Circle y = new Circle( 1, new Point()); // notice the second argument
    Circle z = new Circle(2,a);
    System.out.println("Circle x: " + x);
    System.out.println("Circle y: "+ y);
```

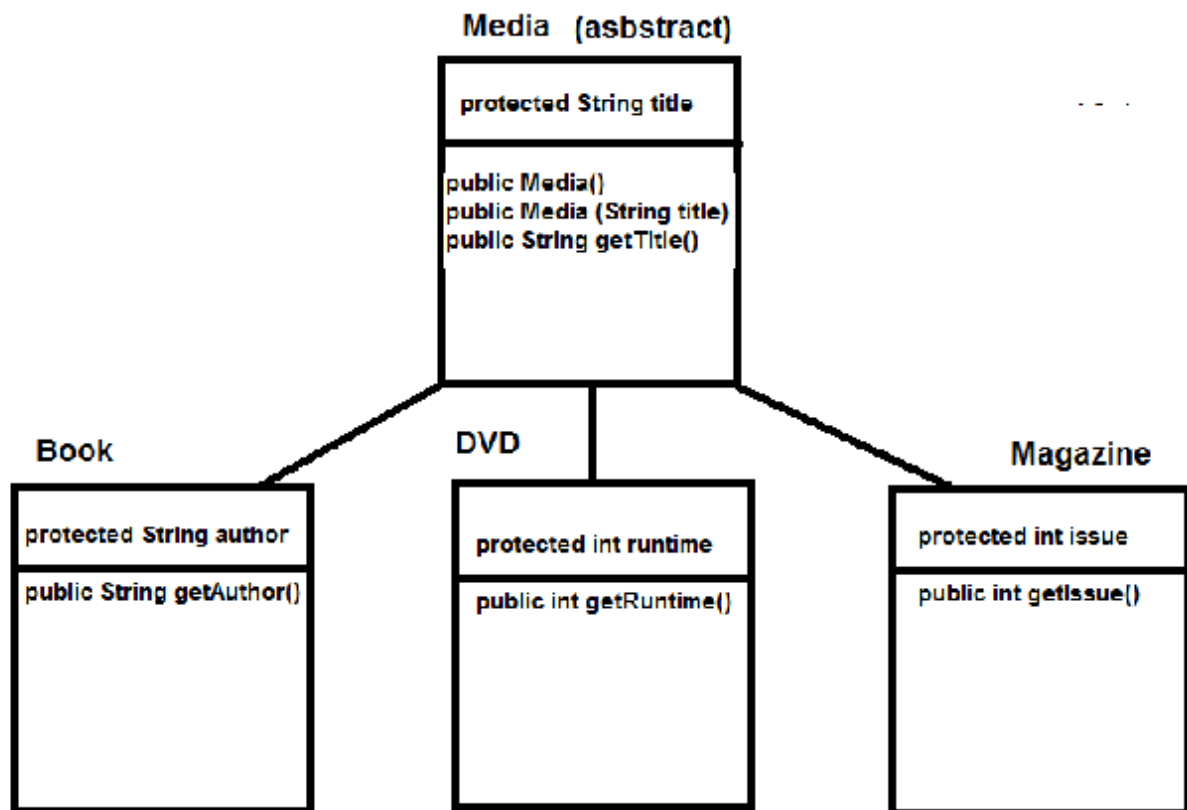
```

    System.out.println("Circle z: "+ z);
    System.out.println("Circle x equals Circle y --> "+ x.equals(y));
    System.out.println("Circle x equals Circle z --> "+ x.equals(z));
}

```

Program 2.

Here is a simple hierarchy. Write classes for each class in the hierarchy



Now make another class **Library** that has a main() method and uses the other classes

The only data in Library is an array **Media[] media**. This array can hold Books, DVDs or Magazines.

The array is initialized in the constructor of **Media**.

In main, the user is asked for an index (in the array)

Then **media[index]** is passed to a method information

```

public void information(Media m)

```

the method `information(...)` prints the title and the author, runtime, or issue depending on the object stored in the array .

DO NOT override `toString()` - even though you could easily do that and should do that. **Think of this as an exercise the use `instanceof` and downcasting.**

Here is a skeleton of Library class.

```
public class Library
{
    private Media [] media;
    public Library() // constructor
    {
        // create a Media array of size 5 and add
        // 2 books , 2 DVDs and 1 magazine
        // here you are using upcasting
    }
    public void information(Media m)
    {
        // fill in the code you will use instanceof here
    }
    public static void main(String[] args)
    {
        // instantiate a Scanner
        //instantiate a Library object

        // prompt the user for an array index 0-4, i

        // loop until user enters 999 for the index
        {
            // pass the Media object at i to information
            // prompt for another array index
        }
    }
}
```

Program 3

Now extend `Media` again and make classes `Book1`, `DVD1`, `Magazine1` each with `toString()`. And make a class `Library1`.

```
public class Library1
{
    private Media [] media;
    public Library1() // constructor
    {
        // as in problem 2
    }
}
```

You do not need the method information(...) for this version.

Use a main method

```
public static void main(String[] args)
{
    // instantiate a Scanner
    //instantiate a Library1 object

    // prompt the user for an array index 0-4

    // loop until user enters 999 for the index
    {
        //print the information for the selected item,
        // prompt for another array index
    }
}
}
```

You do not need the method information(...). Using `toString()` makes it easier and less complex.

