**Lab 1**

(a) Inheritance (1 point)

Design and implement a set of classes that define various [types](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of reading   
material.  
  
Start by defining a parent [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  called ReadingMaterial. This [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  should contain  
the following [instance variables](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) :  
• title - a [String](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  containing the title of the reading material  
• author - a [String](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  containing the author of reading material  
• numPages - an [integer](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  containing the number of pages in the reading material  
All three [variables](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  should be [initialized](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  with [values](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  passed into the   
[constructor](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) .  
  
ReadingMaterial should also have two functions:  
• isLong - takes no [parameters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  and returns a [boolean](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) . If the book has more  
than 250 pages, this function should return [true](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) . Otherwise it returns [false](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) .  
• summary - takes no [parameters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  and returns nothing. This function should  
print out information about the book in the form "[title], written by [author]".  
  
Three classes should inherit from Reading Material: Novel, GraphicNovel, and  
Article.  
  
The Novel [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  should have the following specifications:  
• An [instance variable](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  called [characters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  that contains an [array](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of [Strings](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)   
with the [names](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of the principle [characters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  in the novel (this [variable](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  should  
be [initialized](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  with a [value](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  passed into the [constructor](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) , along with the novel's  
title, author, and number of pages).  
• A function called listCharacters which prints out the [name](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of all the primary  
[characters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  in the order that they are listed, each on their own line.  
  
The Article [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  should have the following specifications:  
• A [String](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  [instance variable](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  called publication that indicates which publication  
the article was published in. This should be [initialized](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  in the [constructor](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) .  
• An overridden version of the summary function that prints out information about  
the article in the folowing form:   
"[title], written by [author], published in [publication]"  
  
The GraphicNovel [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  should inherit directly from the Novel [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  and have  
the following specifications:  
• A [String](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  [instance variable](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  called illustrator that contains the  
[name](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of the person who illustrated the Graphic Novel. (This [variable](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  should  
be [initialized](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  with a [value](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  passed into the [constructor](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) , along with the novel's  
title, author, and number of pages, and main [characters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) .)  
• An overridden version of the summary function that prints out information  
about the book in the following form:   
"[title], written by [author], illustrated by [illustrator]."  
  
  
Test the classes in a Driver [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) . Make one of each [object](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) :  
• A Novel, harryPotter, with the title Harry Potter, author JK Rowling, 303 pages,   
and the main [characters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  Harry Potter, Hermione Granger, Ronald Weasley, and Voldemort.  
• A GraphicNovel, avengers, with the title Avengers, author Stan Lee,  
50 pages, the [characters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  Iron Man, Captain America, Black Widow, The Hulk, and Thor,  
and the illustrator Jack Kirby.  
• An Article, pc, with the title The Social Meaning of the Personal Computer, author  
Bryan Pfaffenberger, 10 pages, and the publication Anthropological Quarterly.  
  
In the Driver [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) , print out the summary of harryPotter, the result of harryPotter's  
isLong function, the list of [characters](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  in avengers, the summary of avengers,   
the result of pc's isLong function, and the summary of pc, with each item on its own   
line.

(b) Polymorphism: (1 point)

Implement the Speaker interface:  
  
public interface Speaker {  
public void speak();  
public void announce([String](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  str);  
}  
  
Then, create three classes that implement Speaker in various ways: 

* SpeakerOfTheHouse: speak [**method**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  prints "I am Speaker of the House.", and   
  the announce [**method**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  takes the [**name**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of a bill in the form of a [**string**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)   
  and prints it in the sentence "The [bill] has passed!"
* SportsAnnouncer: speak [**method**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  prints "Goal!", and the announce [**method**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)   
  takes the [**name**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of a team in the form of a [**string**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  and prints the sentence  
  "The [team] have scored a goal!"
* Actor: speak [**method**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  prints "I've been nominated for three Academy   
  Awards.", and the announce [**method**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  takes the [**name**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of a movie in the form  
  of a [**string**](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  and prints the sentence, "I'm currently staring in [movie]."

Create a driver [class](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) , and in the [main method](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) , prompt the user to enter three  
[Strings](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  -- the [name](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of a bill, the [name](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of a sports team, and the [name](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of a movie.  
Then, create an [object](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  from each of the classes described above and call the speak   
and announce [methods](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  of each [object](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?) , using the [strings](https://pearson.turingscraft.com/codelab/jsp/core_dhtml.jsp?)  provided by the user.

(c) Using Classes in java Library (2 points)

(1) Write a program

* to create a 2x10 two dimensional integer array .
* use Math.random() from java.util.Random to generate 20 integer numbers from 100 to 1000
* store each random number in the two dimension array.
* At the end, print the array. Then switch numbers between two rows then print the array again.

(2) Write a program to accept input string from keyboard and uses substring method to revert

the string and print it out.