

CSCE 156 – Computer Science II

Lab 03 - Classes & Constructors

Sarah Roscoe*

Summer 2020

1 Prior to Lab

Review this laboratory handout prior to lab.

1. Read Java String tutorial:
<http://download.oracle.com/javase/tutorial/java/data/strings.html>
2. Read Java Arrays tutorial:
<http://download.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html>
3. Read Java File I/O tutorial:
<http://download.oracle.com/javase/tutorial/essential/io/file.html>
4. Read the Java tutorial on the `System.printf()` method:
<https://docs.oracle.com/javase/tutorial/java/data/numberformat.html>

2 Lab Objectives & Topics

Following the lab, you should be able to:

- Use Classes and objects to write Java programs
- Understand and use constructors
- Understand the visibility of a class's methods and how to use them

*Developed by Dr. Chris Bourke & Yi Xia

Peer Programming Pair-Up

You may work on this lab with a partner or by yourself. Whichever you choose, you should submit **both your code and the completed worksheet** to the corresponding Handin assignment. The TA will then grade the worksheet for completion.

If you choose to work with a partner, you may work in a *pair programming* setup. One of you will be designated the *driver* and the other the *navigator*. The navigator will be responsible for reading the instructions and telling the driver what to do next. The driver will be in charge of the keyboard and workstation (i.e., shares screen). Both driver and navigator are responsible for suggesting fixes and solutions together. Neither the navigator nor the driver is “in charge.” Beyond your immediate pairing, you are encouraged to help and interact with other pairs in the lab. Each week you should alternate: if you were a driver last week, be a navigator next, etc. Resolve any issues (you were both drivers last week) within your pair. Ask the lab instructor to resolve issues only when you cannot come to a consensus. Whether you complete the lab in pairs or not, it is absolutely essential that you complete any pre-lab activities and familiarize yourself with the handouts prior to coming to lab. Failure to do so will negatively impact your ability to collaborate and work with others which may mean that you will not be able to complete the lab.

3 Getting Started

Clone the project code for this lab from GitHub in Eclipse using the URL, <https://github.com/yi-xia/CSCE156-Lab03-ClassesConstructors.git>. Refer to Lab 01 for instructions on how to clone a project from GitHub.

4 Classes & Constructors

Java is a class-based Object Oriented Programming Language meaning that it realizes the concept of objects by allowing you to define classes which have member methods and variables. Instances of classes are *instantiated* through a constructor; a method with the same name as a class and called using the `new` keyword. This lab will familiarize you with how classes and their constructors are defined and used. In addition, you will be introduced to some ways that Java supports other Object Oriented Principles: Encapsulation and Abstraction.

- *Encapsulation* is the means by which objects group data and the methods/functions that act on that data. It is also the means by which a class's data is protected from the outside world (outside of the object). Java achieves this by allowing you to define member methods and variables and to specify the visibility of these fields

(using the keywords `private`, `protected`, and `public`).

- *Abstraction* refers to the means by which an object exposes a public interface to the outside world while hiding the inner workings (the internal representation or the implementation details). Java's main mechanism for supporting this is the same as with encapsulation though it does provide other means (interfaces, abstract classes).
- *Class Signaling* refers to invoking methods on an instance of a class. Java uses the dot (or period) operator to signal a class. That is, to invoke one of its methods.

5 Activities

We have provided a Java project that simulates a library collections system. It has several classes already defined to model authors, books, a library (a collection of books) and a text-based interface which allows you to search the collection, add books to the collection, and list the collection.

5.1 Constructing Constructors

1. Run the library program to familiarize yourself with its functionality. Note that printing the collection is not fully operational
2. Complete each of the accessor (getter) methods in the `Book` class. Best Practice Tip: always use the `this` keyword to disambiguate the scope of variables and prevent potential problems when subclassing.
3. Observe that the `Book` class does not have a constructor defined. Examine the `addBook()` code and determine how it is possible to create instances of the `Book` class without a constructor.
4. Modify the `Book` class by adding and implementing the following constructor.

```
1  public Book(String title, Author author, String isbn,  
2      String publishDate) {  
3      //TODO: your code here  
4  }
```

5. Adding this constructor will cause syntax errors in other parts of the program. Think about why and then fix these problems by modifying the code appropriately.

5.2 Enforcing Good Encapsulation

The `Book` class is well-designed: it logically groups data and methods together that semantically define what a book is and how you can use it. The `Author` class is not well defined though; its data members are publicly exposed and it has no methods at all.

1. Redesign the `Author` class and make its member variables private.
2. Create and use getter methods to make the members accessible to the outside world. Use these methods where appropriate.
3. Create setter methods (also called mutator methods) to enable code outside of the `Author` class to change the member variables. Add some data validation: for example, do not allow “invalid” values for member variables.
4. Add and make use of an appropriate constructor to this class.
5. Add a method to return a `String` that is the author’s last name/first name separated by a comma and then utilize it where appropriate (modify the `printBooks()` method to use this new method instead of formatting the last name/first name directly).

5.3 Adding and Using Methods

The `printBooks()` method prints out the title, author, and ISBN in a formatted manner one per line. In this activity, you will modify it to output additional information: the year of its publication and its age (the number of years since its publication date).

1. Modify the `printBooks` method in the `LibraryDemo` class to output the publication year of each book in another column. Note that the `Book` class offers a `getPublishDate()` method already implemented for you.
2. To add the “age” column, you will need to add a new method to the `Book` class that returns the number of years between the publish date and today. You may find the following code snippet useful (it utilizes the date/time library provided by Java 8):

```
int years = Period.between(this.publishDate, LocalDate.now()).getYears();
```

5.4 Advanced Activity (Optional)

1. The `printBooks()` method prints books in the order in which they appear in the list. This isn’t as useful as if they were sorted in some manner. Read Oracle’s tutorial on Object ordering, <http://docs.oracle.com/javase/tutorial/>

<collections/interfaces/order.html>. Write code to use the `Collections.sort()` method along with an anonymous `Comparator<Book>` instance to sort the collection according to the book's title.

2. Composition is when a class owns instance(s) of other classes (the `Book` class owns an instance of the `Author` class). If different instances of a class A_1 and A_2 are given references to the same object B , then changes are made to B , the changes will be apparent to both A_1 and A_2 . Sometimes this behavior is desired. Other times it is not. One common method to prevent this is to define a *copy constructor* which takes an instance of the class and performs a *deep copy* of an object by copying each of its fields to the new instance. For example, the copy constructor for the `Author` class may look something like this:

```
1 public Author(Author author) {  
2     this.firstName = new String(author.firstName);  
3     this.lastName = new String(author.lastName);  
4 }
```

Design and implement a copy constructor for the `Book` class and the `Library` class. How do different fields need to be copied?

5.5 Deck of Cards (Optional)

A `PlayingCard` represents a card used in games such as poker and black jack, and stores the suit value (hearts, diamonds, clubs, or spades) and the rank value (2 through 10, or jack, queen, king, or ace). A `Deck` represents a complete 52-card collection of `PlayingCards`. A `MultipleDeck` represents one or more Decks of cards (the exact number is specified in the constructor). Implement the three classes `PlayingCard`, `Deck`, and `MultipleDeck`, providing the minimal functionality to

- shuffle a deck
- deal a card
- check if there are remaining cards.

Make sure to have a `toString()` method to ensure that you can represent a card to the console.

6 Expected output

Note that, sorting is in the Optional activity, so you are only required to have 4 menu options: Add, Find, Print and Exit.

```

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:3

TITLE                                AUTHOR                                ISBN                                PUBLISH YEAR    AGE
War and Peace                        Tolstoy, Leo                         978-0199232765   1869             151
Data Structures & Problem Solving Using Java Weiss, Mark                      0-321-54140-5   2011             9
The Naked & The Dead                 Mailer, Norman                      978-0312265052   1948             72
Barbary Shore                       Mailer, Norman                      0375700390       1951             69
Discrete Mathematics and Its Applications Rosen, Kenneth                    9780073229720   1998             21
The Adventures of Tom Sawyer        Twain, Mark                        9780143039563   1876             144
Adventures of Huckleberry Finn     Twain, Mark                        1885             135
American Gods                      Gaiman, Neil                      0-380-97365-0   2002             18
The Colour of Magic                 Pratchett, Terry                   0-86140-324-X   1983             37
Mort                                Pratchett, Terry                   0-575-04171-4   1987             33
The Hitchhiker's Guide to the Galaxy Adams, Douglas                     978-0345391803   1979             41
Dirk Gently's Holistic Detective Agency Adams, Douglas                     0-671-69267-4   1987             33

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:1
Please enter the details of the book you want to add to the library
Enter the title of the book:
Thinking in Java
Enter the first name of the author:
Bruce
Enter the last name of the author:
Eckel
Enter the ISBN of the book:
978-0131872486
Enter the publication date (YYYY-MM-DD)
2006-02-20
Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:3

TITLE                                AUTHOR                                ISBN                                PUBLISH YEAR    AGE
War and Peace                        Tolstoy, Leo                         978-0199232765   1869             151
Data Structures & Problem Solving Using Java Weiss, Mark                      0-321-54140-5   2011             9
The Naked & The Dead                 Mailer, Norman                      978-0312265052   1948             72
Barbary Shore                       Mailer, Norman                      0375700390       1951             69
Discrete Mathematics and Its Applications Rosen, Kenneth                    9780073229720   1998             21
The Adventures of Tom Sawyer        Twain, Mark                        9780143039563   1876             144
Adventures of Huckleberry Finn     Twain, Mark                        1885             135
American Gods                      Gaiman, Neil                      0-380-97365-0   2002             18
The Colour of Magic                 Pratchett, Terry                   0-86140-324-X   1983             37
Mort                                Pratchett, Terry                   0-575-04171-4   1987             33
The Hitchhiker's Guide to the Galaxy Adams, Douglas                     978-0345391803   1979             41
Dirk Gently's Holistic Detective Agency Adams, Douglas                     0-671-69267-4   1987             33
Thinking in Java                    Eckel, Bruce                       978-0131872486   2006             14

```

Figure 1: Expected outputs (1) for running the library program.

```

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:2
Please enter a Search Option:
  (1) Search By Title (2) Search By Author (3) Keyword Search
1
Enter your search term: Mort

TITLE                                AUTHOR                                ISBN                                PUBLISH YEAR  AGE
Mort                                Pratchett, Terry                     0-575-04171-4    1987          33

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:2
Please enter a Search Option:
  (1) Search By Title (2) Search By Author (3) Keyword Search
2
Enter your search term: Mark

TITLE                                AUTHOR                                ISBN                                PUBLISH YEAR  AGE
Data Structures & Problem Solving Using Java    Weiss, Mark                     0-321-54140-5    2011           9
The Adventures of Tom Sawyer                    Twain, Mark                     9780143039563    1876         144
Adventures of Huckleberry Finn                  Twain, Mark                     1885             135

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:2
Please enter a Search Option:
  (1) Search By Title (2) Search By Author (3) Keyword Search
3
Enter your search term: Java

TITLE                                AUTHOR                                ISBN                                PUBLISH YEAR  AGE
Data Structures & Problem Solving Using Java    Weiss, Mark                     0-321-54140-5    2011           9
Thinking in Java                                Eckel, Bruce                    978-0131872486   2006          14

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:4
Books Sorted!

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:3

TITLE                                AUTHOR                                ISBN                                PUBLISH YEAR  AGE
Adventures of Huckleberry Finn                  Twain, Mark                     1885             135
American Gods                                   Gaiman, Neil                    0-380-97365-0    2002           18
Barbary Shore                                   Mailer, Norman                  0375700390       1951           69
Data Structures & Problem Solving Using Java    Weiss, Mark                     0-321-54140-5    2011           9
Dirk Gently's Holistic Detective Agency         Adams, Douglas                  0-671-69267-4    1987           33
Discrete Mathematics and Its Applications        Rosen, Kenneth                  9780073229720    1998           21
Mort                                              Pratchett, Terry               0-575-04171-4    1987           33
The Adventures of Tom Sawyer                    Twain, Mark                     9780143039563    1876         144
The Colour of Magic                             Pratchett, Terry               0-86140-324-X    1983           37
The Hitchhiker's Guide to the Galaxy            Adams, Douglas                  978-0345391803   1979           41
The Naked & The Dead                             Mailer, Norman                  978-0312265052   1948           72
Thinking in Java                                Eckel, Bruce                    978-0131872486   2006          14
War and Peace                                   Tolstoy, Leo                    978-0199232765   1869          151

Welcome to the Arcadia Library.
Please enter a choice: (1) Add a book, (2) Find a book, (3) Print Collection (4) Sort (5) Exit:5
Thank You for Using Arcadia Library !

```

Figure 2: Expected outputs (2) for running the library program.

```

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: DIAMONDS 12

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
2
Dealing from Multiple Deck:
Multi-Deck Dealing: DIAMONDS 4

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
3
Shuffling single deck.
Deck shuffled!

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
4
Shuffling multiple decks.
Deck shuffled!

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
2
Dealing from Multiple Deck:
Multi-Deck Dealing: CLUBS 10

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: SPADES 11

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: CLUBS 1

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: HEARTS 1

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: CLUBS 2

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: SPADES 8

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: HEARTS 4

Please make a choose: (1) Deal from Single Deck, (2) Deal from Multiple Deck, (3) Shuffle Single Deck, (4) Shuffle Multiple Decks, (5) Exit:
1
Dealing from Single Deck:
Dealing: SPADES 3

```

Figure 3: Expected outputs (3) for running the deck program.