# **WEP2019**: { **Java Programming** }

Homework: **Class and Object**

**Due:**

**Date** dueDate = **new** **Date("**Sunday, 16th December 2017, 11:59pm**");**

**if**(submitDate > dueDate)

totalScore-=20 per day late;

**if**(fileName!="yourName\_CH5\_HW1.zip")

totalScore-=50;

System.out.println("Have Fun and Good Luck!!!");

**The assignment**

1. We would like to model the book as the class **Book**.

The class needs to have the following instance variables to store the “state” of the book

* **title** as String: the book's title
* **author** as String: the book's author
* **publishYear** as int: year of publishing
* **isBorrowed** as boolean: indicates whether or not a book is currently borrowed (checked-out)

The class also needs to implement the following instance methods:

* **a constructor**: takes three arguments: title, author and year of publishing
* **borrow()**: marks a book as borrowed
* **returnBack()**: marks a book as NOT borrowed
* **getTitle()**: returns the title of a book
* **getAuthor()**: returns the author of a book
* **getPublishYear()**: returns the year of publishing
* **getBorrowed()**: returns whether or not a book is currently borrowed
* **toString()**: displays the books author, title , year of publishing and borrowed status in a nice format.

1. Then create a class **Demo** to test your **Book** class by instantiating 10 book objects and add them into an array with the following characteristics:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Object Name | Title | Author | Year of publishing | Is borrowed? |
| book1 | Don’t make me think | Steve Krung | 2000 | Yes |
| book2 | Pro Java Programming | Terrill | 2005 | No |
| book3 | Programming in Python 3 | Mark Summerfield | 2005 | Yes |
| book4 | The Clean Coder | Robert | 2011 | Yes |
| book5 | The Art of Agile Development | James Shore | 2007 | No |
| book6 | Head First Design Patterns | Eric Freeman | 2004 | No |
| book7 | The Linux Command Line | Willian | 2009 | Yes |
| book8 | Code Complete | Steve | 1993 | No |
| book9 | Unit Test Patterns | Gerard Meszaros | 2003 | No |
| book10 | The C++ Programming Language | Bjarne Stroustup | 2013 | Yes |

Try to produce the output as below to console by making use of overriding toString() method:

Title: Don't Make Me Think

by Steve Krung--published 2000

Status: Not available to Borrow

Title: Pro Java Programming

by Terrill--published 2005

Status: Available to Borrow

Title: Programming in Python 3

by Mark Summerfield--published 2005

Status: Not available to Borrow

Title: The Clean Coder

by Robert--published 2011

Status: Not available to Borrow

Title: The Art of Agile Development

by James Shore--published 2007

Status: Available to Borrow

Title: Head First Design Patterns

by Eric Freeman--published 2004

Status: Available to Borrow

Title: The Linux Command Line

by Willian--published 2009

Status: Not available to Borrow

Title: Code Complete

by Steve--published 1993

Status: Available to Borrow

Title: Unit Test Patterns

by Gerard Meszaros--published 2003

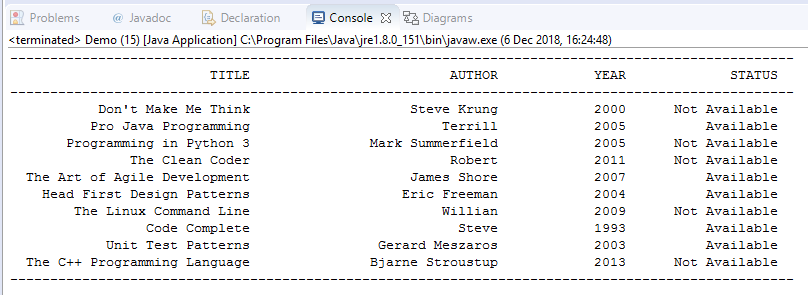
Status: Available to Borrow

Title: The C++ Programming Language

by Bjarne Stroustup--published 2013

Status: Not available to Borrow

Then try to produce the output as below in console.



**Hint:** Try to search about format the output in console java

1. Then we would like to create another class Student. This class model the student who could borrow and return books from the list of books that we had previously. The student could borrow only maximum 3 books and could borrow books which have the status available.

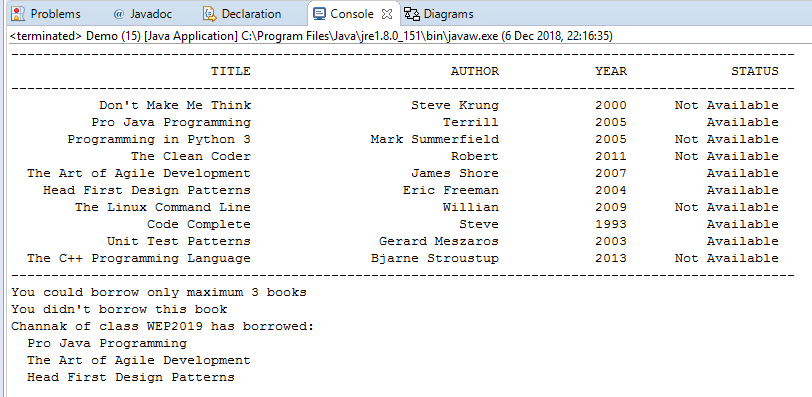
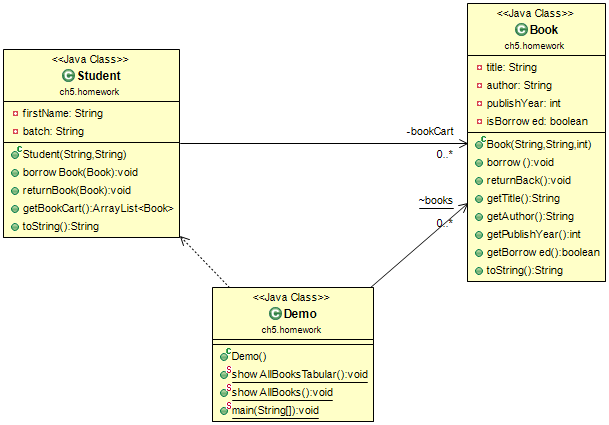
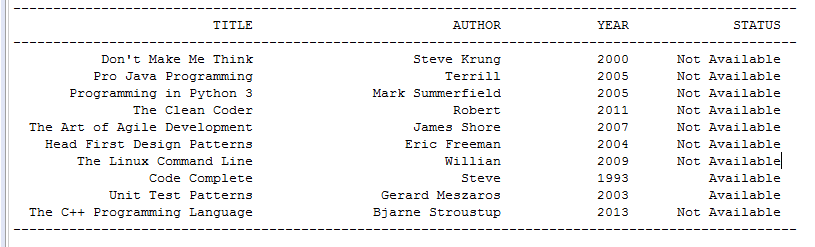
This class has instance variables as below:

* **firstName** as String: name of the student
* **batch** as String: batch of the student
* **bookCart** as ArrayList of book: store the borrowed books. **Hint:** try to understand about ArrayList.

This class also needs to implement the following instance methods:

* a constructor with 2 parameters: first name and batch. We need to initiate the bookCart inside the constructor
* **borrowBook(Book)**: add book to bookCart and mark it as booked if this book is available otherwise output “This book isn't available!””. If the a student tries to borrow more than 3 books output “You could borrow only maximum 3 books”
* **returnBook(Book)**: mark book as not borrowed if the book is in the bookCart otherwise output “You didn't borrow this book”
* **getBookCart()**: returns all the borrowed books
* **toString()**: output the name and batch and all the books the student borrowed

Then test your student class by creating object student with first name as your name and batch of your batch.

When we borrowed the available book after these book’s status changed to Not Avialable

When we try to borrow more than 3 books

When we try to return book that we dint borrowed

