Intro to Java Week 6 Coding Assignment

**Points possible:** 70

|  |  |  |
| --- | --- | --- |
| Category | Criteria | % of Grade |
| Functionality | Does the code work? | 25 |
| Organization | Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear. | 25 |
| Creativity | Student solved the problems presented in the assignment using creativity and out of the box thinking. | 25 |
| Completeness | All requirements of the assignment are complete. | 25 |

**Instructions:** In Eclipse, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week’s assignments and push this document, with your Java project code, to the repository. Add the URL for this week’s repository to this document where instructed and submit this document to your instructor when complete.

**Coding Steps:**

For the final project you will be creating an automated version of the classic card game *WAR.*

Think about how you would build this project and write your plan down. Consider classes such as Card, Deck, and Player and what fields and methods they might each have. You can implement the game however you’d like (i.e. printing to the console, using alert, or some other way). The completed project should, when ran, do the following:

* Deal 26 Cards to two Players from a Deck.
* Iterate through the turns where each Player plays a Card
* The Player who played the higher card is awarded a point
  + Ties result in zero points for either Player
* After all cards have been played, display the score.

Write Unit Tests using Mocha and Chai for each of the functions you write.

**Screenshots of Code:**

**A picture containing text, screenshot, electronics, display

Description automatically generatedText

Description automatically generatedA picture containing text, screenshot, electronics, computer

Description automatically generatedA picture containing text, screenshot, electronics, computer

Description automatically generatedA picture containing text, screenshot, electronics, display

Description automatically generatedA picture containing text, screenshot, electronics, computer

Description automatically generated**

**Screenshots of Running Application:**

**Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated**

**Graphical user interface, application, Word

Description automatically generated**

**URL to GitHub Repository:**

**https://github.com/daniela0723/Week-6**