Course 2 Unit 4 Practice Exercise 11: Arrays

Getting Started – Clone your repository

1. Click on the appropriate link below and accept the assignment to create your repository with starter code and for submitting your work:

    a. **Gallant AM**: https://classroom.github.com/a/-wirC57F
    b. **Gallant PM**: https://classroom.github.com/a/9PD8Lsxj
    c. **Nunn AM**: https://classroom.github.com/a/QAagvxTd
    d. **Nunn PM**: https://classroom.github.com/a/em8b5Weh
    e. **Wijaya AM**: https://classroom.github.com/a/8A-DLhC5
    f. **Wijaya PM**: https://classroom.github.com/a/iMM1LNI4

2. In GitHub Desktop, clone the repository you just created to your desktop.
3. Double-click the index file in the Help folder and click the ConsoleCards link in the pane on the left; this is the documentation for the classes I provided to you in the ConsoleCards dll.
4. Open the project in Visual Studio.

Problem 1 – Create objects

Because you'll be using the classes in the `ConsoleCards` namespace from the dll I added to the project, I added a using directive for that namespace at the top of the Program.cs file.

1. Inside the `Main` method, declare a deck variable and create a new `Deck` object for that variable.
2. Inside the `Main` method, declare an array variable that will hold 5 cards and create a new array object for that variable.
3. Tell the deck to shuffle itself.
4. In GitHub Desktop commit your changes with message: "Completed problem 1".

Problem 2 – Add card to the array, flip it over, and print it

1. Take a card from the top of the deck and add it to element 0 in the array.
2. Flip the card at element 0 of the array over.
3. Tell the card at element 0 of the array to print itself.
4. In GitHub Desktop, commit your changes with message: "Completed problem 2".

Problem 3 – Add another card to the array, flip it over, and print both cards

1. Take a card from the top of the deck and add it to element 1 in the array.
2. Flip the card at element 1 of the array over.
3. Tell the cards at elements 0 and 1 of the array to print themselves.
4. Copy the output of your last test from the console terminal. (Ctrl-A to select all, Ctrl-C to copy)

Submit your work

1. In GitHub Desktop, commit your work with the comment “Ready to Grade” and push to remote.
    a. By committing and pushing to GitHub, you are submitting your assignment to GitHub classroom. If autograding is enabled, it will also test your code and let you know if it’s correct.
2. Return to CodeHS and paste your output to complete the assignment.