Coding Standards

Table of Contents

Introduction

1.1 - Naming Conventions

- Camel Case for functions and variables
- No functions or variables with a length greater than twenty four
- Functions and variables with similar names should be avoided
    - Names for functions and variables should be descriptive
    - Ex: If a function is supposed to change a user's password, it could be:
        - changeUserPass(string oldPass, string newPass)
- Function and variable names should not be too abbreviated
    - Ex: If a function is supposed to send data to a server, it should not be named as such
        - sDS(string data)

1.2 - Style and Spacing

- Development done in Visual Studio Code should have two space tab standard indentation
- Each function and variable should have a comment explaining their purpose when they are first declared
    - This way someone reviewing the code can find the function/ variable declaration and immediately see its description
- Spacing between lines of code when necessary
    - Ex: between two function declarations, one line should be blank
- Excessive spacing should be avoided
    - Ex: do not need three blank lines between code

Contribution Guidelines

2.1 - Function Standards

- If implementing a new function, make sure it follow the previous naming and style conventions
- Function names should be descriptive
    - Ex: addQuestion() for adding a question
- Function names should not be excessively long or too short

2.2 - Comments and Testing

- Make sure to add comments to each significant change you make or any new variables/ functions that you implement
- Comments should be descriptive, but they do not need to be paragraphs long.
- Make sure comments are helpful and only include necessary information
- Tests will be run for each pull request and check in

Code Review Process

3.1 - Checking in Code

- Process for pushing code up to git repository is: git add ., git commit -m “<insert message described the nature and extent of changes>”, git push (to push changes up)
- People working on the code should use their own branch instead of pushing directly main so that code must go through a review process
- Pushing code up to a branch should occur every time a significant change is made. Changes do not always mean complete fixing of a problem, but a certain extent of progress that was made

3.2 - Making Pull Requests

- Make a pull request when you have fully fixed an issue
    - If you make a pull request when an issue is only half fixed, then there is no point of putting any of the code in main
    - Making pull requests too early will slow down the development process
- Pull requests should have a description
    - Should describe what the original issue was, and what was causing it
    - Methods taken to solve the issue
    - Any blocks the developer ran into while fixing the issue
- Only make pull requests comparing your branch to main
- Make sure to run tests to see if the issue is fixed before making a pull request