

# Practice Assignment: W09\_10-26\_1

---

## CISC 3120 Section ER6 Design and Implementation of Software Applications I

This exercise helps you review input/output streams, and JavaFX UI design and implementation. The instructor wrote a simple HTML editor application called `SimpleEditor`. You may browse the code on Github via the following URL,

<https://github.com/CISC3120/sampleprograms/tree/master/fileio/SimpleEditor/>

The instructor has left some code out in the application. Your tasks are to complete it. Below are the tasks.

1. Copy the `SimpleEditor` project to your weekly practice repository.
2. Read the code and understand the design of the application. In particular, pay attention to the comments in the `SimpleEditor` class about how you may introduce some formalism into application design.
3. In the `SimpleEditor` class, you will find the following method. The implementation of the method is not given. Implement the method that reads the content of the file represented by `theFile` object where `theFile` is an instance variable of the class.

In the `files` folder in the project, you will find the `brooklynhomepage.html` file. You may use this file for testing purpose.

```
1 public String readFile() throws FileNotFoundException , IOException
2     // TODO: task 1
3 }
```

4. In the `SimpleEditor` class, you will find the following method. The implementation of the method is also not given. Implement the method that writes the String value of the `htmlText` variable to the file referenced by `theFile`.

```
1 public void saveTheFile(String htmlText) throws FileNotFoundException , IOException {
2     // TODO: task 2
3 }
```

*If you implement the above two methods correctly, the application should function, i.e., you can create a new file, open an existing one for editing, and save the file you created or edited. You should test your implementation and see if works.*

5. In addition, you should notice that the application has a `Help|About MenuItem`. However, it does not response to users' action simply because no code has been written for it. You will provide the functionality for the `MenuItem`, and the functionality is that the application displays a short description of your choice to describe the application when a user clicks on it.

The instructor suggests that you follow these steps.

- (a) The application is designed with a *Model-View-Controller* pattern. You shall add a `MenuItem` to the `MainSceneController` class, and annotate it with `@FXML`. This is to “*inject*” the `MenuItem` control from the `fxml_mainscene.fxml` view to the controller class.
- (b) Following the above, you will add the `fx:id` attribute whose value is the variable name of the `MenuItem` control you added in the above to the corresponding `MenuItem` in the `fxml_mainscene.fxml` view.
- (c) You can now implements the `onAction` handler for the `MenuItem` in the controller class.
- (d) Finally, you should revise the `fxml_mainscene.fxml` view so that the `onAction` attribute of the `MenuItem` is set to the `onAction` handler in the controller class.

As discussed in the class, you may directly edit the `fxml_mainscene.fxml` file in Eclipse or edit it using the `JavaFX Scene Builder`.