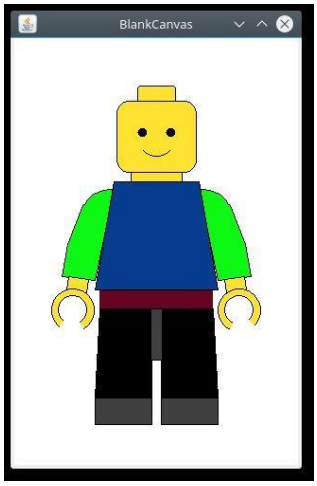
Activity 3.1 - MiniFig (Part 1)

Overview

In this activity, you will write a program that *uses* a provided `MiniFig` class to create your own Avatar.



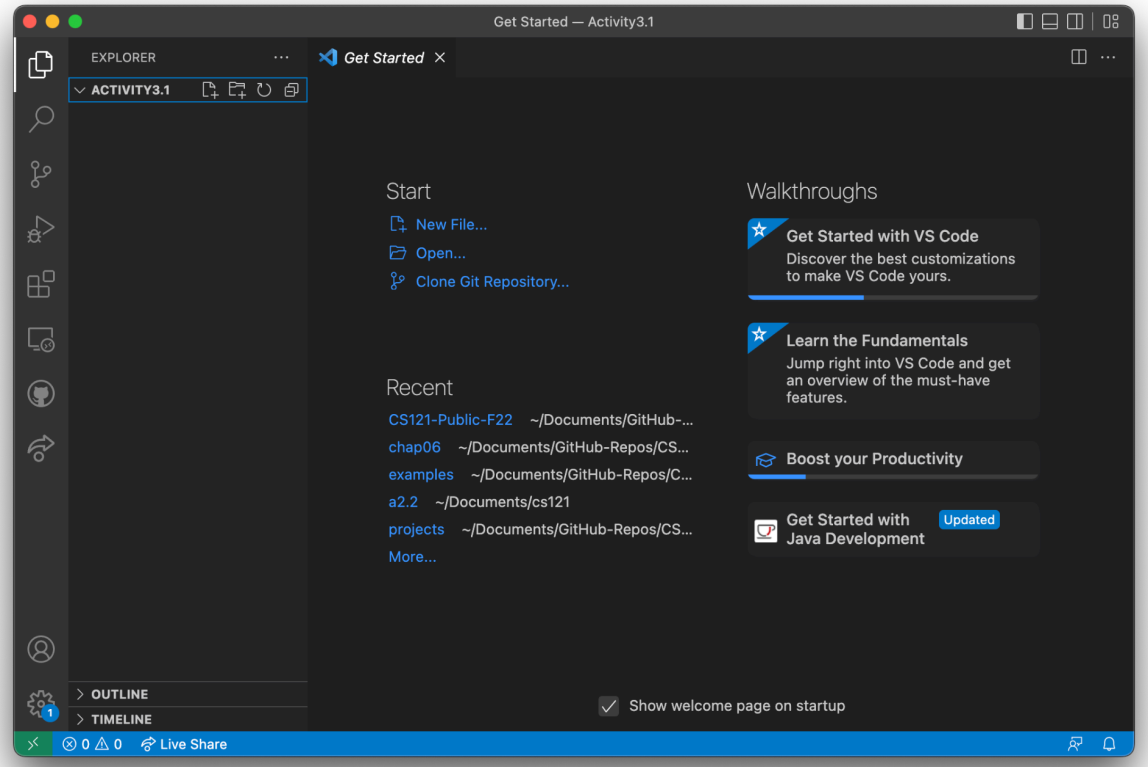
The MiniFig class provides several methods for customizing colors and retrieving attributes of the MiniFig object.

Part 1 Instructions

Getting Started

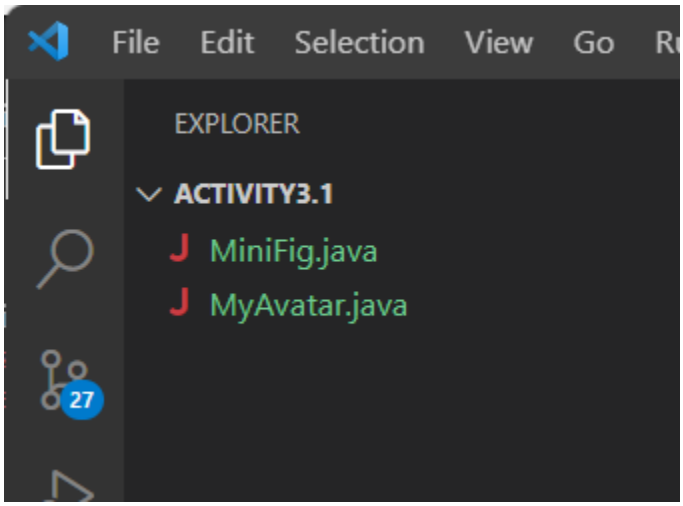
1. Open VS Code. To be sure you are starting fresh, select `File → Close Folder` (If there is no "Close Folder" in the menu, that means there is no folder currently open and you can proceed to step 2.)
2. Create a new folder in your cs121 directory. Select `File → Open Folder`, and navigate to your cs121 directory and create a new folder inside of it named "Activity3.1" and select that new "Activity3.1" folder.

3. When you finish this step, your Activity3.1 folder should be the only thing open (see figure below, note the “Explorer” panel only has Activity 3.1).



4. Download two files: `MiniFig.java` and `MyAvatar.java` (Click on the links, then right click and use *Save as* option. Make sure to save as files with *.java* extension as some systems will add a *.txt* extension).
   For Windows users, if file extensions are hidden:
   a. Open **File** Explorer; if you do not have an icon for this in the task bar; click Start, click **Windows** System, and then **File** Explorer.
   b. Click the **View** tab in **File** Explorer.
   c. Click the box next to **File** name **extensions** to **see file extensions**.
5. Import `MiniFig.java`. You will not modify this class, you will just use it. You can import a file by saving it in the desired location. In this case, you will save MiniFig.java in your `cs121/Activity3.1/` directory.
6. Import `MyAvatar.java` by saving it in your `cs121/Activity3.1/` directory.

7. Now ready your VS Code window to work on this specific activity. Select `File → Close Folder`. This will close the cs121 folder. Now select `File → Open Folder` and choose your cs121/Activity3.1 folder. Your explorer panel should like like the one below:



8. Complete the class according to the specifications below.

Specifications

Use constructors, setter methods, and getter methods

For each of the following steps, follow the directions in the TODO comments listed in MyAvatar.java.

1. Instantiate (create by calling the constructor) a new `Point` object.
2. Instantiate a new `MiniFig` object. Here is the documentation for the MiniFig class.
3. Instantiate a new `Color` object.
4. Invoke (call) the `setTorsoColor` method on your MiniFig object. This is an example of a setter method.
5. Invoke the `draw` method on your MiniFig object.
6. Invoke the `getBaseMidPoint` method on your MiniFig object. This is an example of a getter method.

Create an Alias

1. Create an alias for your MiniFig object and change the torso color of your alias. What happens when you draw your original MiniFig?

```
MiniFig bob = new MiniFig(g, scaleFactor, anchor);
```

```
bob.setTorsoColor(Color.GREEN);

MiniFig robert = bob;
robert.setTorsoColor(Color.RED);

bob.draw();
// or
// robert.draw();
```

Exploring the MiniFig API

If time permits, try customizing your avatar further using the methods of the MiniFig class.

Terminology Identification

In your code add comments identifying examples of the following: constructor, attribute, method, instantiate an object, reference variable. These should be identified with an @keyterm tag within the comment.

Code Review

When you are finished with this activity, pair up with a classmate and show off your MiniFig!

Submission

After completing the assignment, use the assignment link in Canvas and follow the submission instructions there. You will upload your `MyAvatar.java` file and submit your reflection in the "Comments" box.

Reflection Requirements

Write a one paragraph reflection describing your experience with this activity. The reflection should also include the name of your code review partner AND something interesting you found in their code. Please review the activity rubric for more details.