Tutorial 14 – GitHub Desktop

Contents

[1 Setup 1](#_Toc73628429)

[1.1 Google Account 1](#_Toc73628430)

[1.2 Download and install Git 1](#_Toc73628431)

[1.3 Signup for a GitHub account 1](#_Toc73628432)

[1.4 Install GitHub Desktop 1](#_Toc73628433)

[2 Using GitHub Desktop 1](#_Toc73628434)

[2.1 Example 2](#_Toc73628435)

This tutorial was prepared after previous ones and several steps are marked that you have already completed them. This is another way to manage integrating your local work with GitHub. It uses GitHub Desktop. I have investigated this a number of years ago, and it wasn’t so friendly. Now, it is fine. I present this tutorial so that you have yet another option for managing your work flow.

# Setup

## Google Account

**[Omit, unless you haven’t done this]** Obtain a Google account if needed and make note of the email address.

## Download and install Git

**[Omit, unless you haven’t done this]** Download and install Git. I opted in for “Windows Explorer Integration,” though I don’t think we will use this. I accepted all other defaults.

<https://git-scm.com/downloads>

## Signup for a GitHub account

**[Omit, unless you haven’t done this]** Go to pricing page, choose the Free account. Use your gmail account.

<https://docs.github.com/en/github/getting-started-with-github/signing-up-for-github/signing-up-for-a-new-github-account>

## Install GitHub Desktop

Use the link below to download and install GitHub Desktop.

<https://desktop.github.com/>

At the conclusion of installation, it should launch. Use your GitHub username to connect GetHub Desktop (GHD) to your GH account. Also, specify your gmail address (it defaults to a *noreply* address). When complete, you’ll see a *Let’s get Started* dialogs.

If you have problems, you can reference:

<https://docs.github.com/en/desktop/installing-and-configuring-github-desktop/installing-and-authenticating-to-github-desktop/installing-github-desktop>

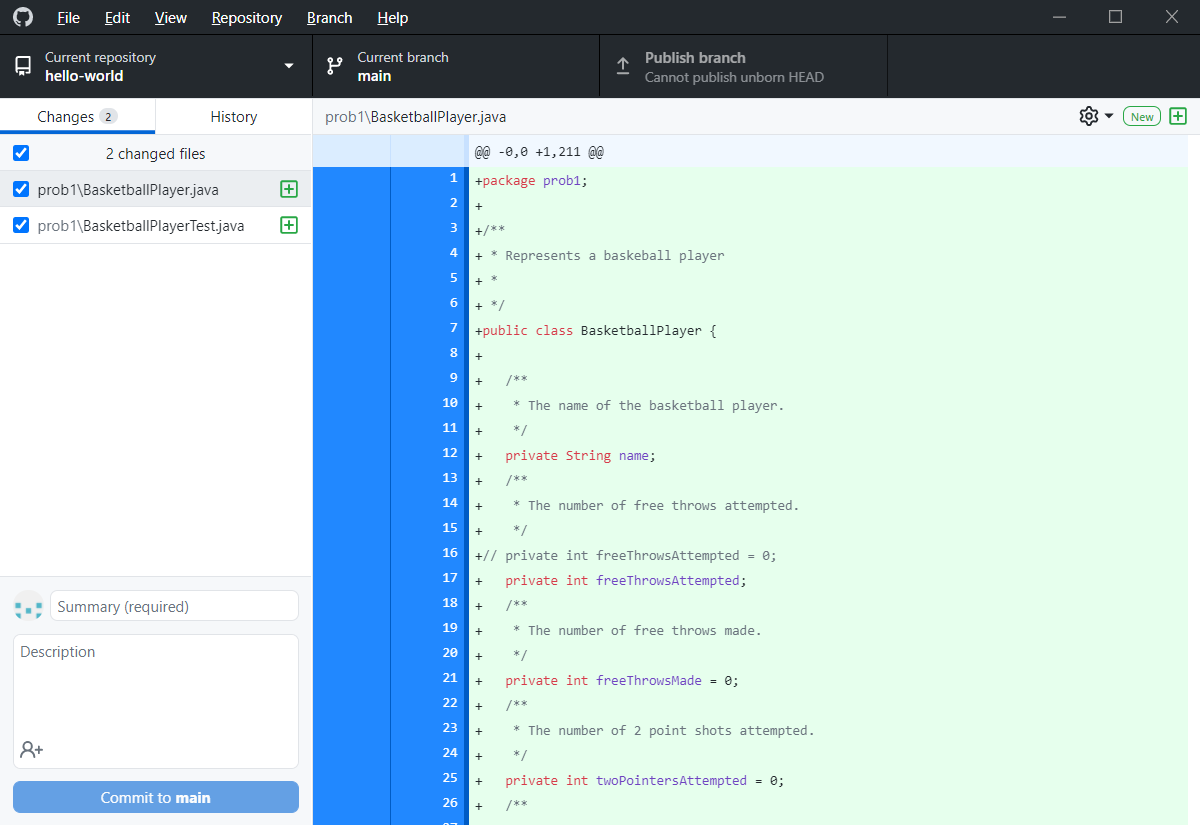
# Using GitHub Desktop

## Example

1. Download and unzip the code from the Schedule, *bball.zip.* Remember the path to the *prob1* folder; you will need it shortly.
2. Follow these steps:

<https://docs.github.com/en/github/creating-cloning-and-archiving-repositories/creating-a-repository-on-github/creating-a-new-repository>

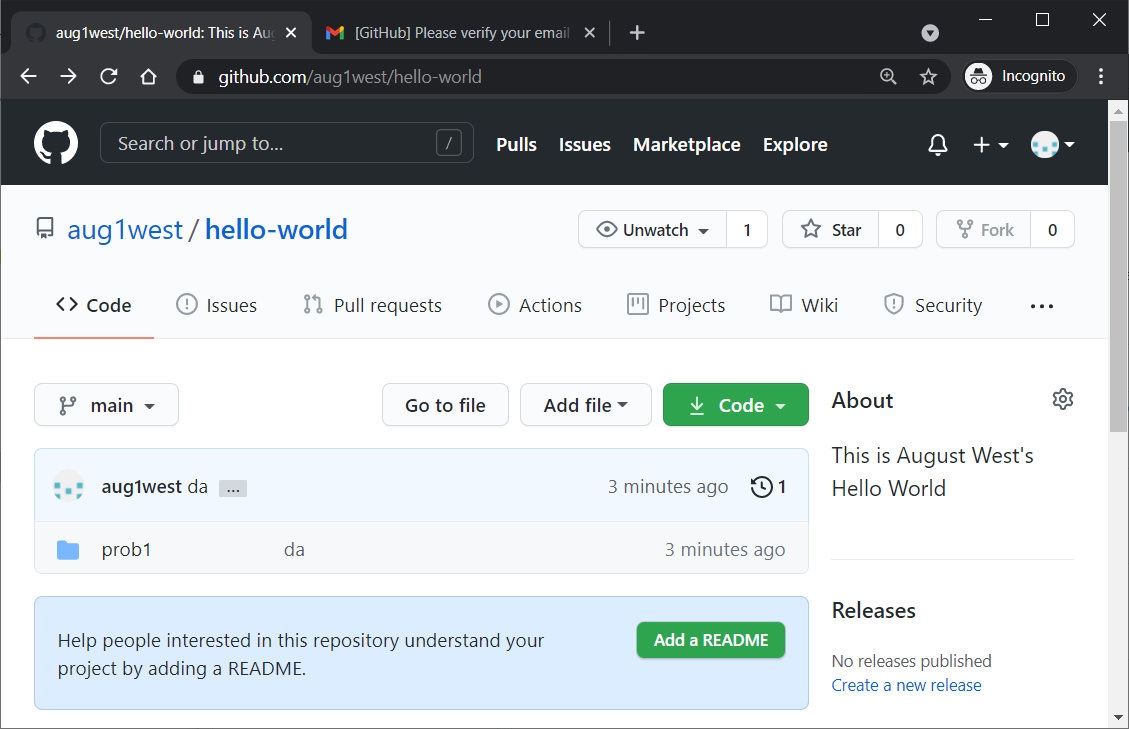
1. At the end, step 10, in GitHub, towards the top, in the blue highlighted Quick Setup section choose: *Set up in Desktop.* This should open GHD.
2. Accept the defaults and choose: Clone
3. Choose: Show in Folder. This is your GitHub repository which has been *cloned* onto your harddrive. There is nothing in it (except a *.git* folder).
4. Copy the *prob1* folder from above into the folder from the previous step (*i.e.* copy *prob1* into your repository)
5. Return to GHD. The window will look like this:



1. Type a Summary:

Initial submission of basketball code

1. Choose: Commit to main
2. Choose: Publish Branch
3. Go to your GH page (in the web browser) and you will see the *prob1* folder



1. Drill down into *prob1* and you will see the two files. Open one by clicking.
2. Edit the file by choose the Pencil icon.
3. Make a small change to the first line of documentation by adding “ZZZ” to the end (so we can identify the change later)

package prob1;

/\*\*

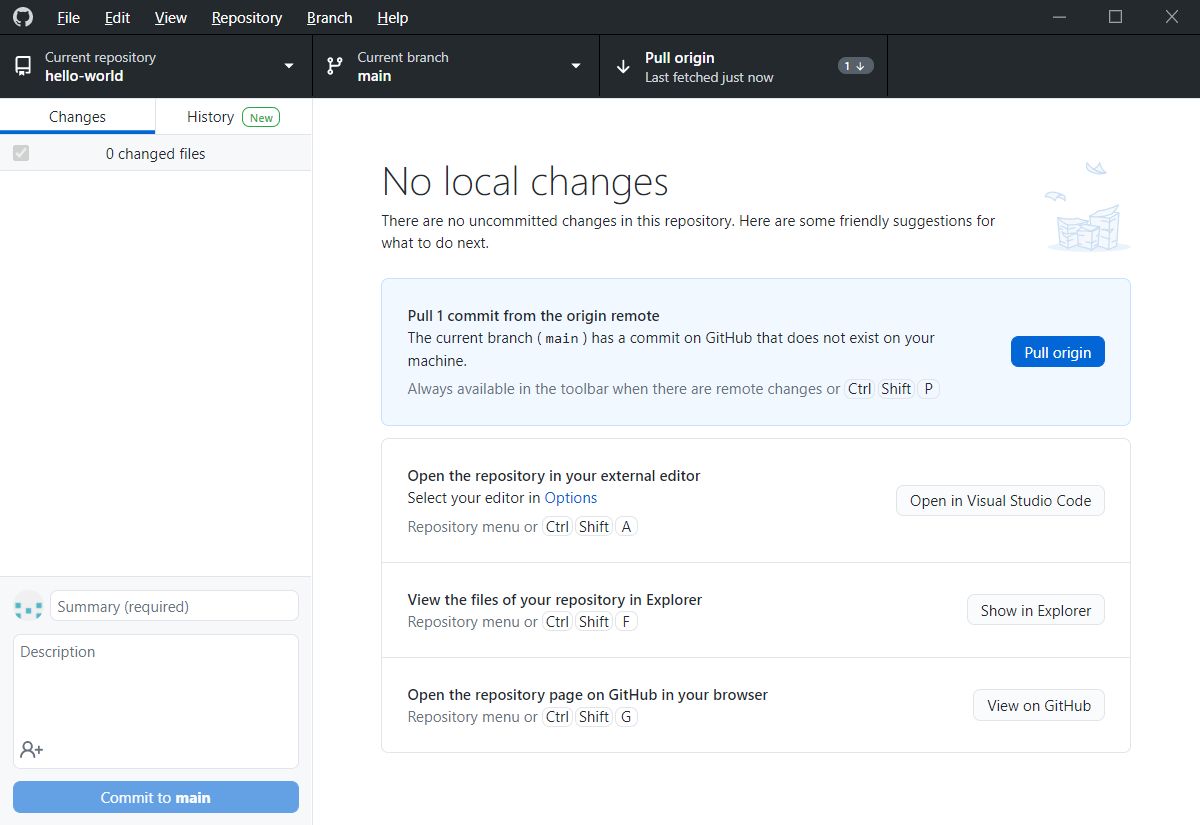
\* Represents a baskeball playerZZZ

\*

\*/

public class BasketballPlayer {

1. Scroll to the bottom. You can type a commit message, but one is already supplied (which is an unacceptable one, but we will use it as default for now). Choose: Commit changes.
2. Return to GHD. Choose, from the top: Fetch origin. The window will look like this:



1. Choose: Pull origin. This will pull the changes to your local repository.
2. In GHD, choose: Repository, Show in Explorer.
3. Drill down to the file you change on GitHub and verify that it is changed in your computer.
4. **Do the following:**
5. As in Step 19 above, display your file that was just changed.
6. Make a screen shot similar to the one show below making sure it shows your username and the change.
7. Place the image in the *HW VCS* document in the appropriate place.
8. The image should easily readable without zooming in or out.

