Hello!

Welcome to the short Java tutorial I made just for you :)

In this document, you will be introduced to:

1) How to use github at a basic level

**Github:**

\* "Github is a website where people flex their ego" - Jarvis Johnson

Github is a website where you can share your code with other people, WORK with other people, all ONLINE

1 or more people can work on a singular project together without having to worry that much :)  
  
How can people work together without even having to see each other? It’s simple.

Commits, Pushing, and Pulling.

**Commits:**



Commits are changes that someone has made to the file. If you change one line of code and save it, chances are you HAVE to commit it to push it online.

Commit != Saving your code, but Committing is saving your saves (if that makes sense)

**Pushing:**

Pushing simply means moving ALL your commits online. Every commit you make on the files will be pushed online

**Pulling:**

Receiving all the commits that people have pushed. When pushing, if 2 or more people have changed the same line of code, you will get an error. This is known as a **Git Conflict**

|  |  |
| --- | --- |
| Git Tutorial | Commands And Operations In Git | Edureka | Let us go over this image.  Localrepo = The place where YOU are working offline. (Like ATOM)  Remote Repo = The repository that you push to (Like Domino-Game)  The steps are:   * **Add** your code (Make changes) * **Commit** them to your local repository * **Push** them to the Remote Repository * **Pull** any changes someone could have made * **Merge** what you have pulled to your branch (More details later!) |

In summary:

**Change → Commit → Push online → Pull from remote → Merge to Local Branch**

As of now, you created a clone of the online repository, meaning you **already have a local repository set up** 😊

Next, we are going to learn how to commit and push online

Head on over to Domino.java and find a comment that says “Secret Password:”



Add any phrase you’d like and save (CTRL + S) your changes.

Now let’s commit your changes in the simplest way.

Press CTRL + # to open the commit menu

|  |  |
| --- | --- |
|  | Here are ALL the files you saved (Ignore READMEFIRST). Click on all the files you want to commit and click the green + on top. (This is called **staging**)  Once you “staged” all the files you want, simply write a **commit message** summarizing what you have done and click commit! |

Now that you have committed your first file, it’s now time to push your changes online! :)

If you have permission to edit the remote repository, then you will be allowed to push your commits onto the remote repository.



Right-click your project (Domino-Game, in this case) and find:

**Team > Push Branch “master”**

After clicking push, the rest is self-explanatory! 😊

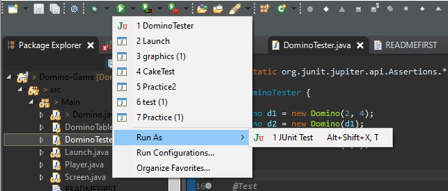
|  |  |
| --- | --- |
|  | Here you can see all the commits you have made on what branches.  If all the commits are correct and you think everything is ready to go, simply click push and done!  **\*\*\*NEVER PUSH BROKEN/ERRONEOUS CODE TO MASTER UNLESS SPECIFICALLY TOLD TO\*\*\*** |

Congratulations! You just pushed your first code onto Github!

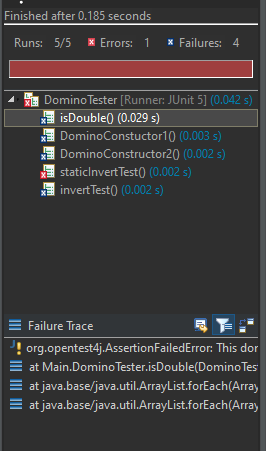
Usually at this point you would want to pull the code that other people have pushed, but in this case, there is nothing else to push until further notice

If you feel confident, head on over to **Domino.java** and finish the exercises that I have provided for you 😊

To test your code, simply run **DominoTester.java** as a Junit file like this:



And the following should pop up:



A blue X means that the code got **the wrong result**

A red X means that the code had an **error and should be fixed**

Double click the test that you have failed, and it will take you to the test case that you need to fix.

When you have solved them all, **try pushing it to Github**!

**Good Luck!**