CS1050 Technical Documentation

Use headings to organize topics and create a table of contents to be able to quickly access information.

[Overview of Developer Technical Documentation](#_e7hnk3gmh67o)

[Set Up Development Environment](#_efvt7vjb7dhk)

[Version Control with Git and Github](#_p4s5uluouzyw)

[IDE Information](#_xda8plfuejto)

[General Resources](#_j8bsu2nohj85)

[Module 1](#_qn332jwlduxf)

[Compilation process](#_8u1ro2aazjxd)

[Variables, Constants, and Primitive Data Types](#_v362y395npbk)

[Arithmetic Operators and Combined Assignment Operators](#_ekcaf49ktnl3)

[Memory Allocation, Primitive Data Types, Conversion and Casting](#_t5qvme324700)

[Quality Code: Naming Conventions](#_2gp23qqtt32h)

[Module 2](#_qke7paye0rm)

[Classes, Objects and Methods](#_41t39wfjzewk)

Follow the 4 tips from [Part 2: Developer Technical Documentation](https://docs.google.com/document/d/1Ve-3OD9EN9DCufeGZTyH_vIt9bsDzUMUv_p8KI1CXlQ/edit#heading=h.4lr6hykc4ng6). This can be done in many ways

* Organize use headings and subheadings
* Summarize, list, screenshots
* Links to resources
* Copy code snippets with comments in your code explaining
* Link to code examples in your repository
* Other resources such as websites and videos. Just remember not all information is good information.

# Overview of Developer Technical Documentation

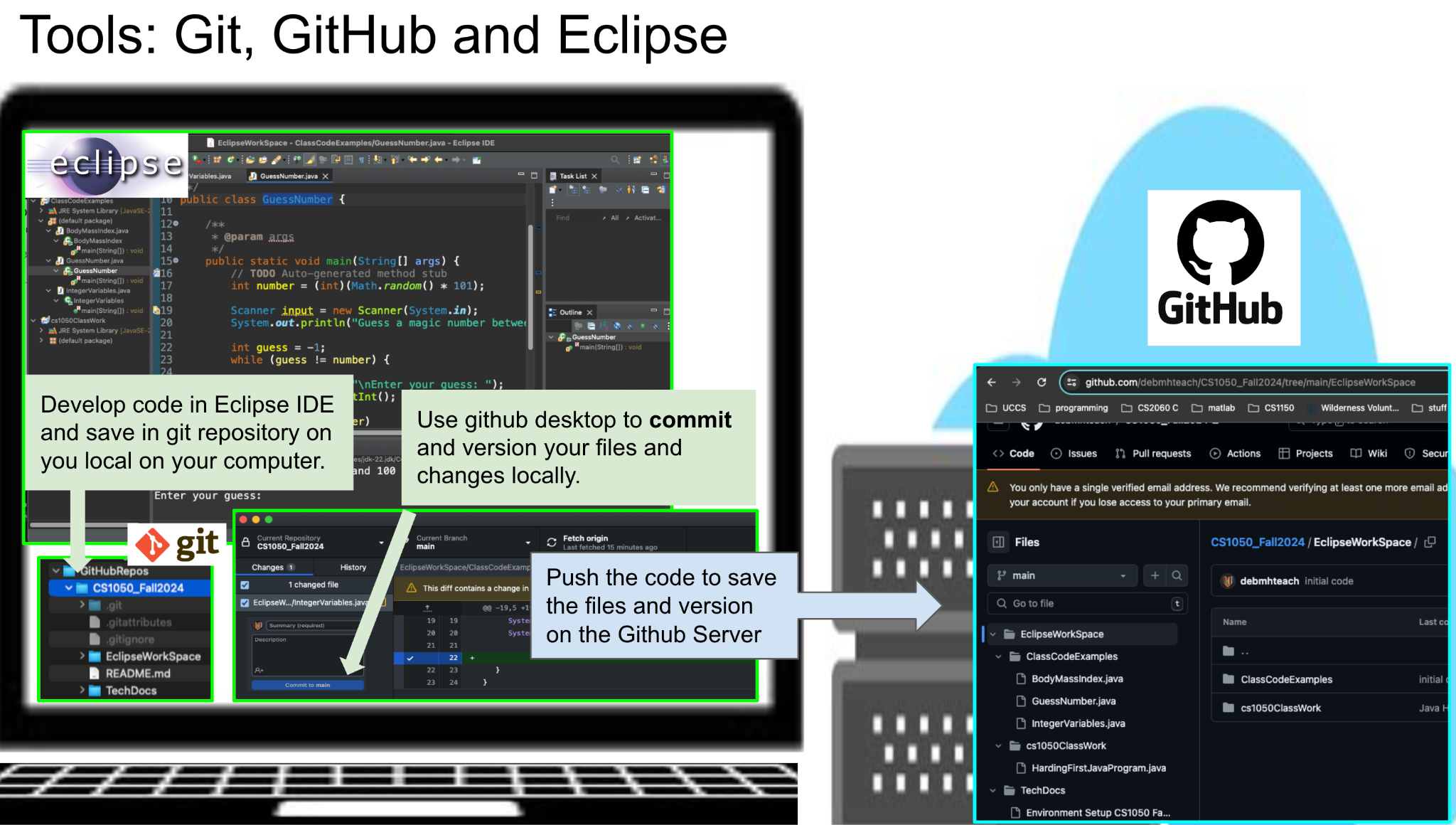
I use headings to help organize and easily access information. I like to include snippets of code that contains comments as an easier way to have documentation. You can put in the document what is helpful for you.

This is your documentation to organize according to your needs. It should include at least the following but can contain more.

# Set Up Development Environment

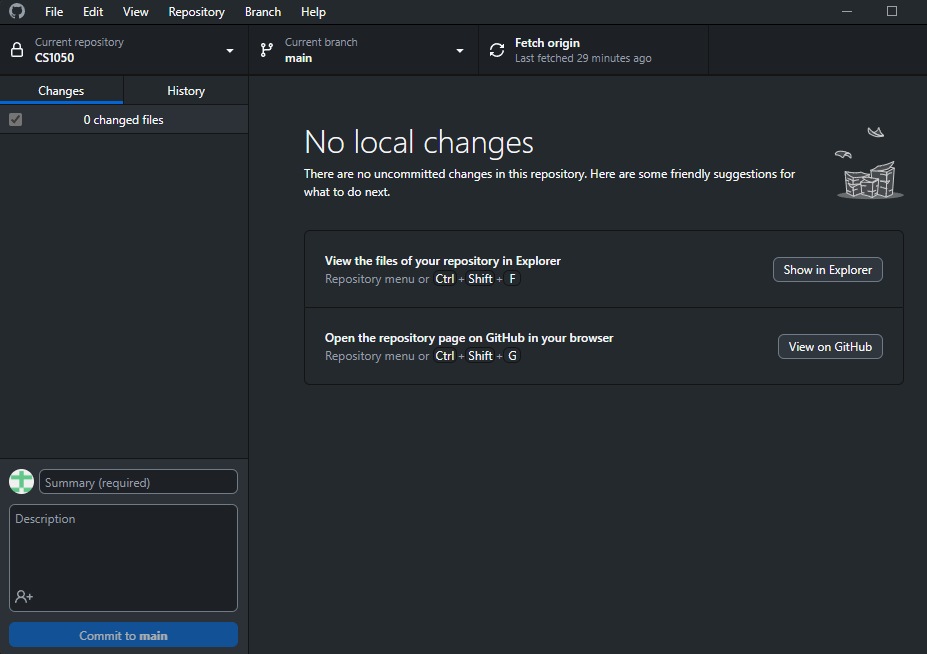
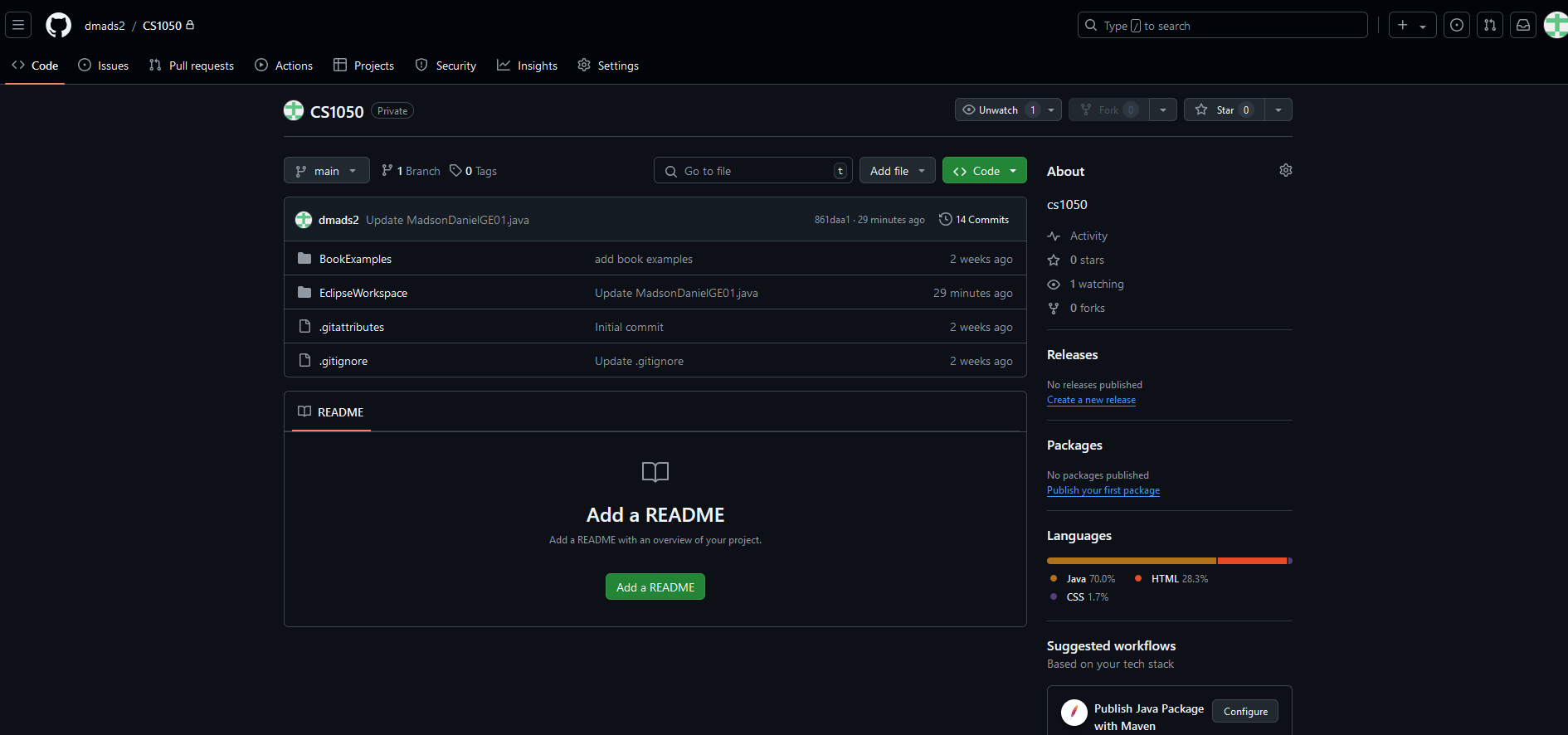
Include what is necessary to set up your environment again. It can be a link to other documents or resources but it is required to have a brief summary of what information is provided in the link.

[Environment Setup CS1050 Fall 2024.docx](https://docs.google.com/document/d/1lnLZdxusq1UNoUshdf3sK64QaTrNF5Iy/edit#heading=h.gjdgxs)



## Version Control with Git and Github

<https://docs.google.com/document/d/1lnLZdxusq1UNoUshdf3sK64QaTrNF5Iy/edit#heading=h.gjdgxs> contains information on how to set up git and github.

* Git-
  + Git is basically a versioning software for your local PC
  + 
* Github-
  + Github is git but online. Some benefits include versioning, online backups, code review, and collaboration.
  + 

## IDE Information

Setting up a project with IDE

<https://docs.google.com/document/d/1lnLZdxusq1UNoUshdf3sK64QaTrNF5Iy/edit#heading=h.gjdgxs> contains information on how to set up Eclipse, create a project and how to create a class..

Is there something you want to put here that you will be doing a lot from the documentation?

<https://dzone.com/articles/how-add-existing-files-eclipse>

HOW TO ADD FILES TO ECLIPSE

CREATING A NEW CLASS

1. Select the appropriate project folder you want the class in
2. Source folder – should be the project name
   1. Package- empty
   2. Name- name for file starts with capital letter
   3. Which method stubs?
      1. Check: public static void main (String[]args)
      2. Uncheck: inherited abstract methods
3. check generate comments
4. done

# General Resources

Here you can list resources that you use frequently. You can add more.

* [Shared student resources containing resources, lectures and assignments](https://drive.google.com/drive/folders/1HvYY8zzSwlsH--03olvqOJooGnJkZ7F4)
* [Draft Schedule](https://docs.google.com/spreadsheets/d/1igBbmOBTXfvEVicyAggnqRIpV5Fwqh64/edit?gid=2047083326#gid=2047083326)
* [Link to join lecture in teams](https://teams.microsoft.com/l/meetup-join/19%3aklQhREluFbWiaroMMZPBYeNPhZa9AFGnTb7ATIPTUFE1%40thread.tacv2/1724008042961?context=%7b%22Tid%22%3a%2203309ca4-1733-4af9-a73c-f18cc841325c%22%2c%22Oid%22%3a%2233eb6fec-88d5-4bc1-bb67-32063f1cfacc%22%7d)

## 

# Module 1

Rather than typing information

* Copy and paste code examples that contain comments
* Summarize concepts
* Include information form lectures
* Links to tutorials that help you.
* Link to code examples in your repository
* Screenshots
* Other resources such as websites and videos. Just remember not all information is good information.
* Use information from your guided exploration

## Compilation process



* Bug icon- debugger
* Green and white arrow- run

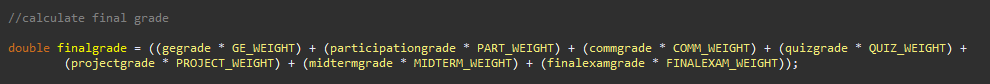


* Termination button- Red square
* Console- where the code gets output

## Variables, Constants, and Primitive Data Types

### 

## Arithmetic Operators and Combined Assignment Operators



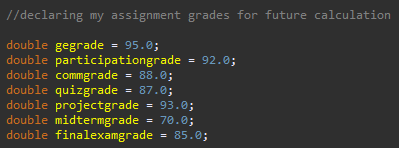
## Memory Allocation, Primitive Data Types, Conversion and Casting

<https://www.javatpoint.com/java-data-types> This website describes primitive data types

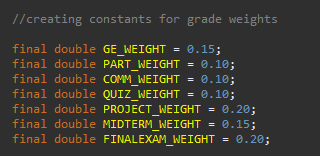
* boolean data type- true/false
* byte data type-
* char data type
* short data type
* int data type- 1,2,3
* long data type
* float data type- 10.0
* double data type- 10.000000000000000000000000000000

## Quality Code: Naming Conventions

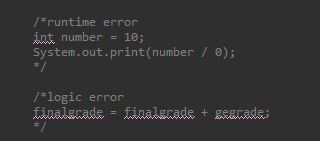
Variables- lowercase and one word



Constants- uppercase and underscores



**Errors:**

****

# Module 2

## Classes, Objects and Methods