

CSE 11: Accelerated Introduction to Programming

Assignment 1

Getting Started with CSE 11

Due: Wednesday, April 10, 11:59 PM

Learning goals:

- Certify your commencement of academic activity in CSE 11
- Configure a remote Linux environment
- Affirm your commitment to upholding academic integrity in CSE 11
- Learn basic CLI commands that manipulate files/directories.
- Install Java 11
- Install the Visual Studio Code
- Write and compile a simple Java program
- Submit files to Gradescope

NOTE: This programming assignment must be done individually.

Part 0: Prior knowledge + certification of commencement of academic activity (0 points)

For every course at UC San Diego, per the US Department of Education, we are **required** to certify whether students have commenced academic activity for a class to count towards eligibility for Title IV federal financial aid. This certification must be completed during the first two weeks of instruction.

For CSE 11, this requirement will be fulfilled via an **ungraded** prior knowledge quiz, which will assist the instructional team by providing information about your background prior to the course.

Complete this quiz in [Canvas](#) by going to the CSE 11 course and navigating to Quizzes. Then click "[First Day Survey: Prior Knowledge #FinAid](#)" and complete the quiz.

Part 1: Setting up UCSD Linux Cloud (0 points, optional)

This part is necessary if you prefer to write your programming assignments in a remote Linux environment with JDK 11 already installed. **Even if you prefer to configure a personal computer for Java development (instructions below), you should become familiar with UCSD Linux Cloud in case you encounter issues with a personal computer.**

UCSD Linux Cloud

You will be accessing a remote instance of the Linux operating system from your regular web browser (e.g., Chrome, Safari, Edge). It also means you can access your personal Linux Cloud environment regardless of which computer you are working with!

Account

You will be using your UCSD personal account to log into Linux Cloud. Use your UCSD username and password.

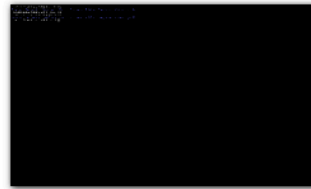
Duo Registration

All accounts must be registered in Duo. If your account is not registered, then register it [here](#).

Linux Cloud

Access [Linux Cloud](#) by logging in with your UCSD personal account credentials. It will then request Duo authentication on your other device. Next, open the remote Linux Desktop by selecting "ieng6 Server Remote Desktops."

RECENT CONNECTIONS

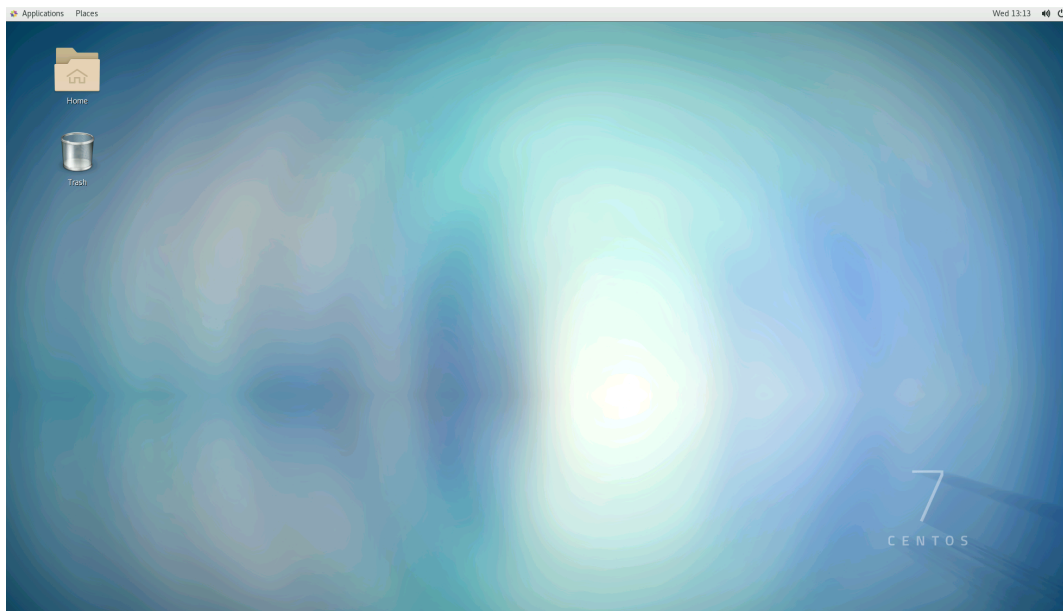


ieng6-240.ucsd.edu SSH

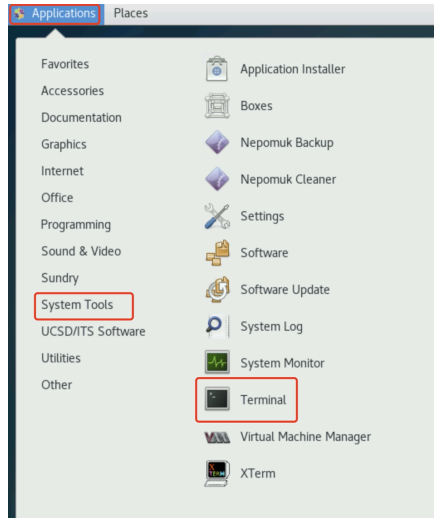
ALL CONNECTIONS

+ ieng6 Server Remote Desktops

Now, you should have a remote Linux Desktop (it may take some time to load) in your web browser that looks similar to this:



To prepare the environment, open the terminal. You can find it under Applications → System Tools → Terminal.



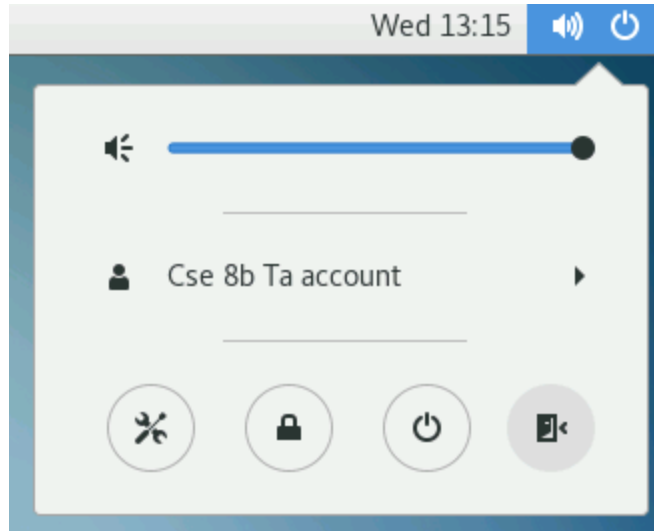
When you log in with your account, you should see a prompt like this:

To begin work for one of your courses [cs11sp24], type its name at the command prompt. (For example, "cs11sp24", without the quotes).

This command is an automatically generated alias that has been set up for students enrolled in the course. If you don't see this message, you can create it manually by doing the following in the terminal:

```
cd /home/linux/ieng6/cs11sp24/${USER} ;  
module load /home/linux/ieng6/cs11sp24/public/modulefiles/cs11sp24;  
cd .
```

When you are finished using Linux Cloud, please ensure you **log out** of your remote desktop environment, **DO NOT just close your browser** window or tab. To log out, click the power button in the top right-hand corner. Then, select the fourth button with the open door.



After you press it, it will prompt you to log off. If you do not log off, Linux Cloud may not let you back in when you later want or need to!

Part 2: Academic Integrity (0 points)

The CSE department supports the fundamental values of academic integrity: honesty, respect, responsibility, fairness, trustworthiness, and courage. Unlike courses from most other departments, you are required to submit code, and there are specific programming qualifications you need to know before you write any code. Most of the time, students violate academic integrity just because they don't know it is a violation. The below tutorial is your one-stop shop of academic integrity to avoid such situations.

To complete Part 1 of this assignment:

1. Carefully read both the collaboration policy and academic integrity policy for this course, which is found in the syllabus.
2. Please watch the [Source Code Plagiarism tutorial](#) thoroughly.
3. After you finish watching the tutorial, you must complete the [CSE Integrity of Scholarship Agreement](#). By signing this, you acknowledge that you have completed and understood the tutorial as well as the agreement. **Although this part of the assignment is worth 0 points, if you do not complete the CSE Integrity of Scholarship Agreement, then you will receive an F in the course.**

Part 3: Command-Line Navigation (0 points)

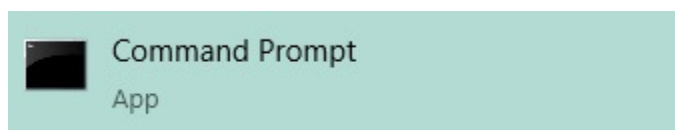
Although this part of the assignment is worth 0 points, basic command-line navigation is required in CSE 11 and related questions may appear on the midterm and final assessments.

Now, let us introduce you to the command line interface (CLI)! You might have seen programmers using a black window and typing commands on it. That is the command-line interface, and every programmer uses it. Throughout this quarter, you will be using the command-line interface to navigate, compile, and run your program for all assignments. It has many names, such as **cmd**, **CLI**, **console**, or **terminal**.

Open the CLI

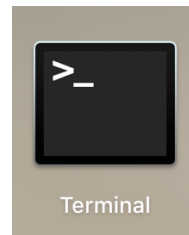
First, let's open the CLI.

- **Windows:** Depending on your version of Windows and your keyboard, one of the following should open a command window (you may have to experiment a bit, but you don't have to try all of these suggestions):
 - Go to the Start menu or screen, and enter "Command Prompt" in the search field.
 - Go to the Start menu → Windows System → Command Prompt.
 - Go to the Start menu → All Programs → Accessories → Command Prompt.
 - Go to the Start screen, hover your mouse in the lower-left corner of the screen, and click the down arrow that appears (on a touch screen, instead flick up from the bottom of the screen). The Apps page should open. Click on Command Prompt in the Windows System section.
 - Hold the special Windows key on your keyboard and press the "X" key. Choose "Command Prompt" from the pop-up menu.
 - Hold the Windows key and press the "R" key to get a "Run" window. Type "cmd" in the box, and click the OK key.

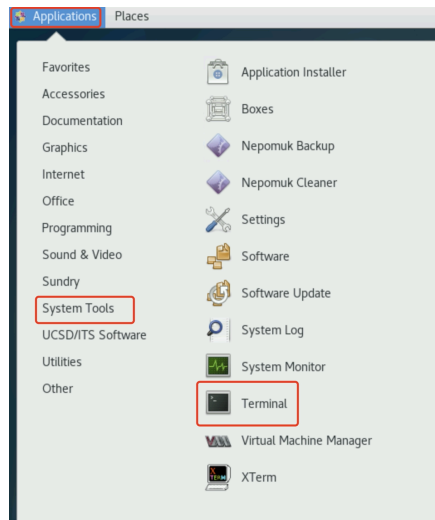


- You may also use Windows PowerShell instead of Command Prompt.
- **macOS:**
 - Go to Applications → Utilities → Terminal.

- Or you can Search “Terminal” in the Spotlight Search bar.



- Linux: We're using [UCSD Linux Cloud](#) as an example. You can find it under Applications → System Tools → Terminal. (Picture next page for your reference)



Command Prompt

Now you should see a black or white window that is waiting for your command.

- Linux and macOS : you probably will see a `$` or `%` on your current line.
- Windows: you probably will see a `>` on your current line.

Also, there might be something like `C:\Users\name` or `name@MacBook` before the above characters. This is totally fine.

IMPORTANT: Each of the commands below will be typed to the right of the command prompt. Do not type command prompt `$` , `%` , or `>` before the command! The command prompt is not part of the command!

Your first command

Now let's type the following command, no matter which OS you are using. Keep in mind that you should only type the command, not the command prompt before it.

- Linux and macOS:

```
$ whoami
```

- Windows:

```
> whoami
```

Then hit enter. And you can see your username appear.

- Linux and macOS:

```
$ whoami  
myusername
```

- Windows:

```
> whoami  
computername\myusername
```

Current directory

If you want to know where you are right now, or the full path of your current directory, you can try the following commands.

- Windows PowerShell, Linux and macOS:

```
$ pwd  
/home/linux/ieng6/cs11sp24/myusername
```

“pwd” stands for “print working directory”. The above result is from UCSD Linux Cloud.

- Windows Command Prompt:

```
> cd  
C:\Users\myusername
```


“cd” stands for “change directory”. The above result is from a Windows machine.

List files and directory

If you want to know what is in your current directory, you can try the following commands.

- Linux and macOS:

```
$ ls
Desktop  Downloads  Pictures  Templates  ...
```

- Windows:

```
> dir
Directory of C:\Users\myusername
04/01/2023 07:28 PM <DIR>      Applications
04/01/2023 07:28 PM <DIR>      Desktop
04/01/2023 07:28 PM <DIR>      Downloads
04/01/2023 07:28 PM <DIR>      Music
...
```

Change current directory

Now let's go to our Desktop directory.

- Linux and macOS:

```
$ cd Desktop
```

- Windows:

```
> cd Desktop
```

If you want to check whether you have successfully changed the directory, use the “pwd” or “cd” command we mentioned above to check your current directory after changing.

Create directory

It would be nice if you can create a CSE 11 directory on your desktop for file management.

- Linux and macOS:

```
$ mkdir cse11
```

- Windows:

```
> mkdir cse11
```

If you want to check whether you have successfully created the directory, use the “ls” or “dir” command we mentioned above to check your current directory’s content.

Practice time!

Based on the above tutorial, please create a new directory named `to_be_deleted` inside of `cse11`, then change your current directory to the `to_be_deleted` directory. Refer to the above commands: “mkdir” to create a directory and “cd” to change directory.

Create a Java file

Now that we’re inside the `to_be_deleted` directory, we want to create an empty java file named `test.java`. Here’s what we should do.

- Linux and macOS:

```
$ touch test.java
```

- Windows Command Prompt:

```
> type nul > test.java
```

- Windows PowerShell:

```
> $null > test.java
```

If you want to check whether you have successfully created the file, use the “ls” or “dir” command we mentioned above to check your current directory’s content.

Delete a file/directory

Before we delete anything, let’s try to go back to the `cse11` directory.

- Linux and macOS:

```
$ cd ..
```

- Windows:

```
> cd ..
```

Using `..` with the `cd` command will change your current directory to the parent directory. That is, the directory that contains your current directory. You can check where you are now by one command we mentioned above.

Now we can delete the directory and its contents all at once.

- Windows PowerShell, Linux and macOS:

```
$ rm -r to_be_deleted
```

The `-r` represents recursive. The “remove recursively” operation is especially for removing directories. It will remove everything inside of the `to_be_deleted` directory.

- Windows Command Prompt:

```
> rmdir /S to_be_deleted
```

`/S`: This tells the `rmdir` command to operate recursively. It means that if the target directory contains subdirectories, those subdirectories and their contents will also be deleted. Similar to `-r` above, it will remove everything inside of the `to_be_deleted` directory.

You can check whether the directory is removed by one command we mentioned above.

However, all remove commands like `del`, `rm`, `rmdir` are irreversible. Think twice before you do any deleting!

How to get help?

One last thing before you go to the next part, we want you to know how to call out the manual for each command. Let's say we want to learn more about the `cd` command.

- Linux and macOS:

```
$ man cd
```

- Windows Command Prompt:

```
> cd /?
```

- Windows PowerShell:

```
> cd -?
```

Help is also available online for [Linux](#) and [Windows](#).


Part 4: Compile and run some code (100 points)

The last part of this assignment is for you to compile and run your first piece of code in CSE 11. If you're using UCSD Linux Cloud or you are developing on a machine in the CSE basement, then JDK 11 is already installed and configured on your CSE 11 class account so you don't need to worry about the installation. However, if you prefer to work on a personal computer, you can find the installation steps below for Windows and macOS.

Install Java 11

1. Navigate to the following link:
<https://www.oracle.com/java/technologies/downloads/#java11>
2. Depending on your OS, select one of the following:

Linux **macOS** **Solaris** **Windows**

- a. 
3. Installation steps for **Windows**:

a. Choose the **x64 Installer**

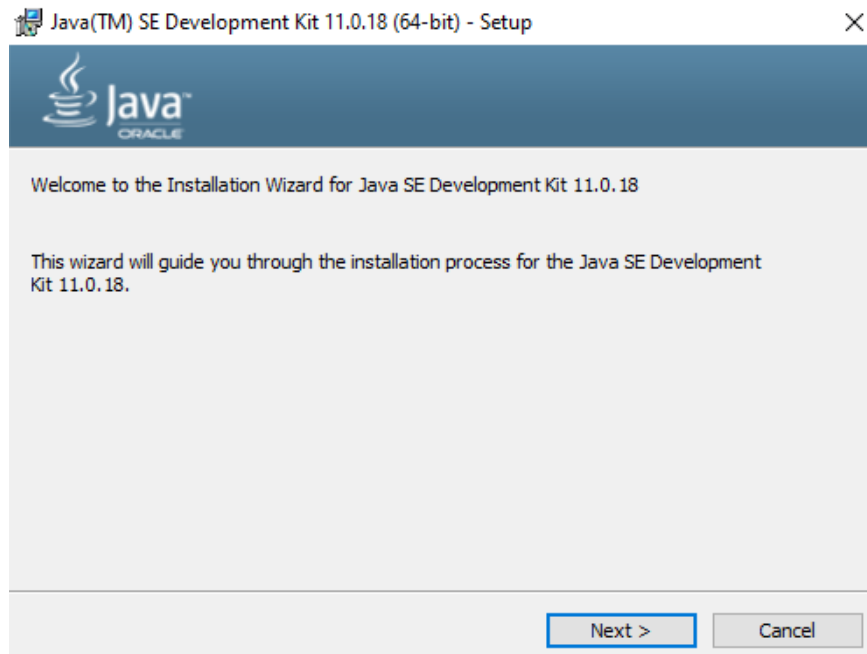
Linux	macOS	Solaris	Windows
Product/file description	File size	Download	
x64 Installer	140.55 MB	jdk-11.0.16.1_windows-x64_bin.exe	

b. It will prompt you to create an Oracle account. Create one, login, and download the executable.

c. If you receive a password validation error even though your passwords match, try creating the account in an Incognito/Private/InPrivate window.

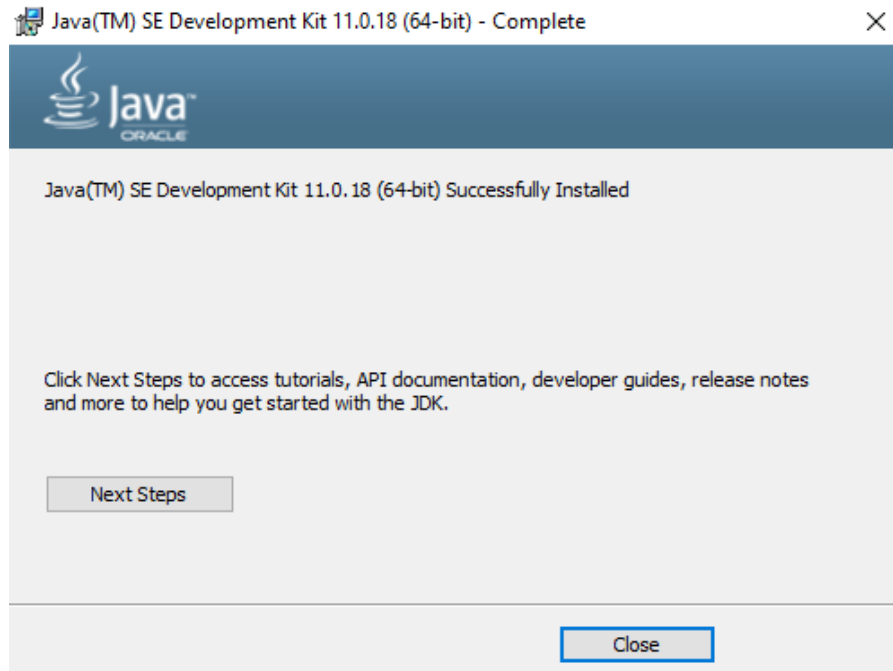
Retype password*
Please address Password validation error

d. Run the executable that you just downloaded and follow through the setup wizard



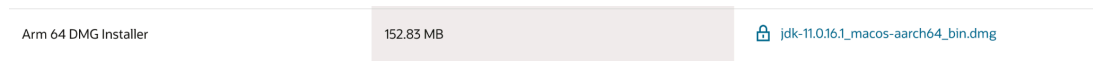
e. Install it in the default directory

- f. You should now have this. You can click close:

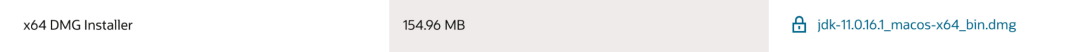


4. Installation steps for **MacOS**

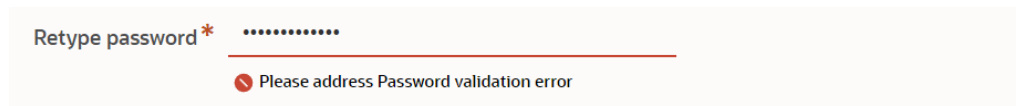
- a. If you have a Mac with Apple Silicon, choose the **Arm DMG Installer**



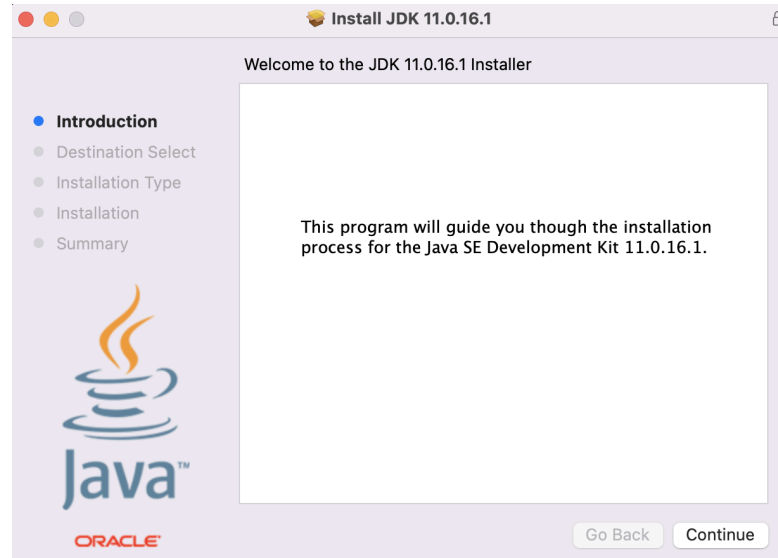
- b. If you have a Mac with an Intel chip, choose **x64 Installer**



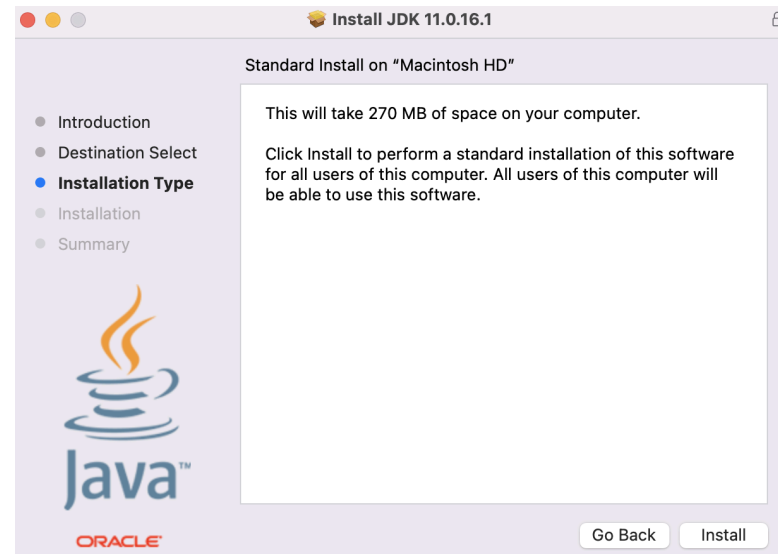
- c. It will prompt you to create an Oracle account. Create one, login, and download the executable.
- d. If you receive a password validation error even though your passwords match, try creating the account in an Incognito/Private/InPrivate window.



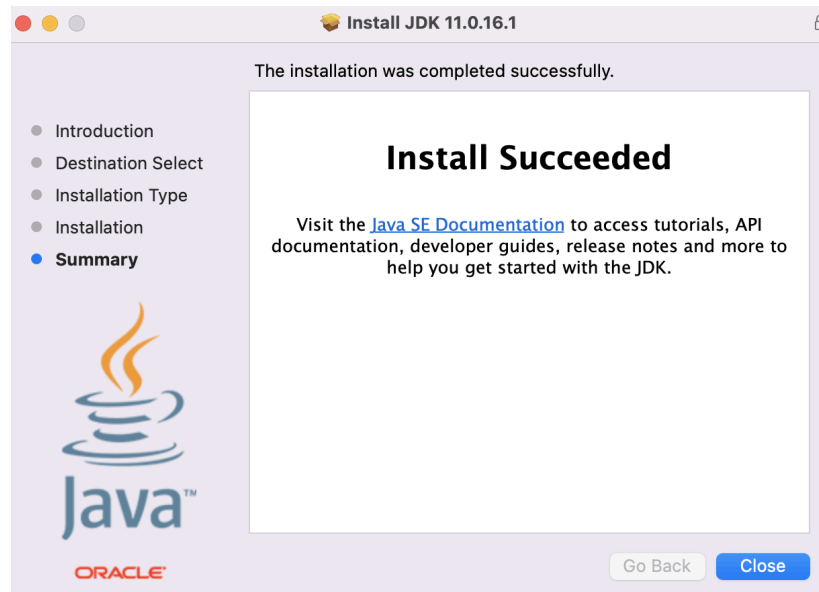
- e. Open the pkg installer and follow through the installation instructions



f. Click "Continue" and you should see this:



g. Provide your login credentials and click "Install". After installation, you should see the following:



h. Click “close” and you’re done!

Note: it’s perfectly fine to see version “11.0.22” in the above screenshots.

Now you should have installed the right version of Java (i.e., Java 11) for this course... but how do you check if you installed it correctly?

Please open your terminal and go to the cse11 directory first (do you remember where you created it?)

Then, run the following command.

- Linux and macOS:

```
$ java -version
```

- Windows:

```
> java -version
```

You should see the Java version "11.0.22" now. We want to make sure you’ve installed Java correctly, so we’re going to ask you to turn in a record of your installation.

Run the following command:

- Linux and macOS:

```
$ java -version 2> java_version.txt
```

- Windows:

```
> java -version 2> java_version.txt
```

You will find a new .txt file generated. Please check if that file contains the correct Java version information. **The file `java_version.txt` will be submitted to Gradescope.** As a sanity check, it should look a little like this:

```
[(base) Marthas-MBP-2:Desktop marthagah]$ cd cse11/
[(base) Marthas-MBP-2:cse11 marthagah]$ pwd
/Users/marthagah/Desktop/cse11
[(base) Marthas-MBP-2:cse11 marthagah]$ java -version 2> java_version.txt
[(base) Marthas-MBP-2:cse11 marthagah]$ ls
java_version.txt
[(base) Marthas-MBP-2:cse11 marthagah]$ █
```

Notice the path when we use the `pwd` (Linux) or `cd` (Windows) command? We're inside the `cse11` directory when we run the command to generate our java version into a txt file. Afterwards, when we use the `ls` (Linux) or `dir` (Windows) command, we can see our file!

Note: If you run into issues or observe a different version number when running these commands after you've installed JDK 11, you may either need to:

1. Restart your terminal and run the commands again
2. Make changes to your system Path variable (though this is typically not necessary):

<https://www.java.com/en/download/help/path.html>

Now that we have Java 11 installed, let's try to compile and run a piece of code from the command line.

Visual Studio Code

You can use any text editor (e.g., Vim, Notepad (Windows), TextEdit (macOS)) you want to edit the files in your programming assignments. Do not use an Integrated Development Environment (IDE) that automatically generates source code, as 1) you need to learn how to write all source code yourself and 2) the resulting source code often will fail to compile and/or run in Gradescope, resulting in a score of 0! Here, we will be installing Visual Studio Code (VS Code), a lightweight, great text editor. Visual Studio Code is already installed in your Linux Cloud environment. However, if you are working with your own machine on Windows or Mac, you can install VS Code by following the link below.

1. Navigate to this link: <https://code.visualstudio.com/download>
2. Download the respective executable:
 - a. For Windows:



- b. For macOS:

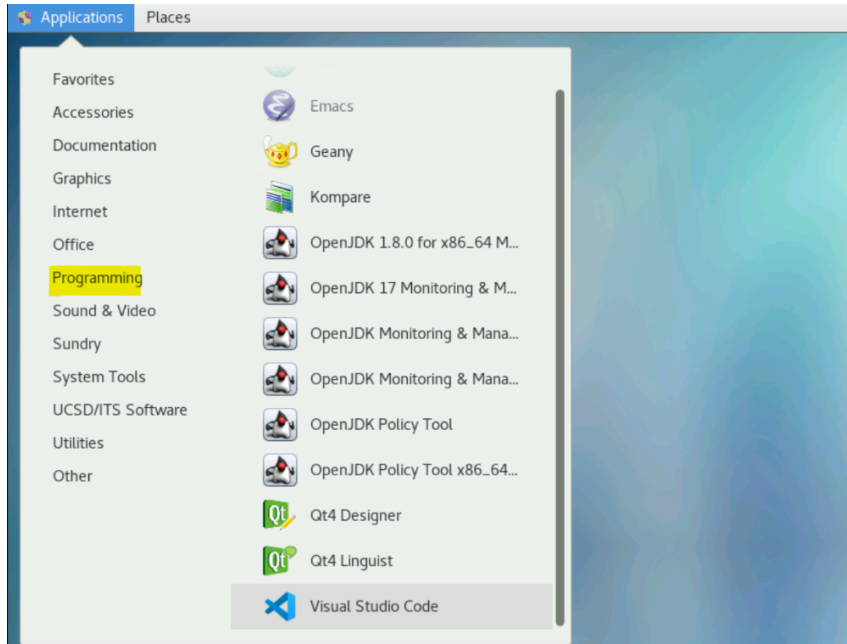


3. Follow the installation instructions depending on your OS.

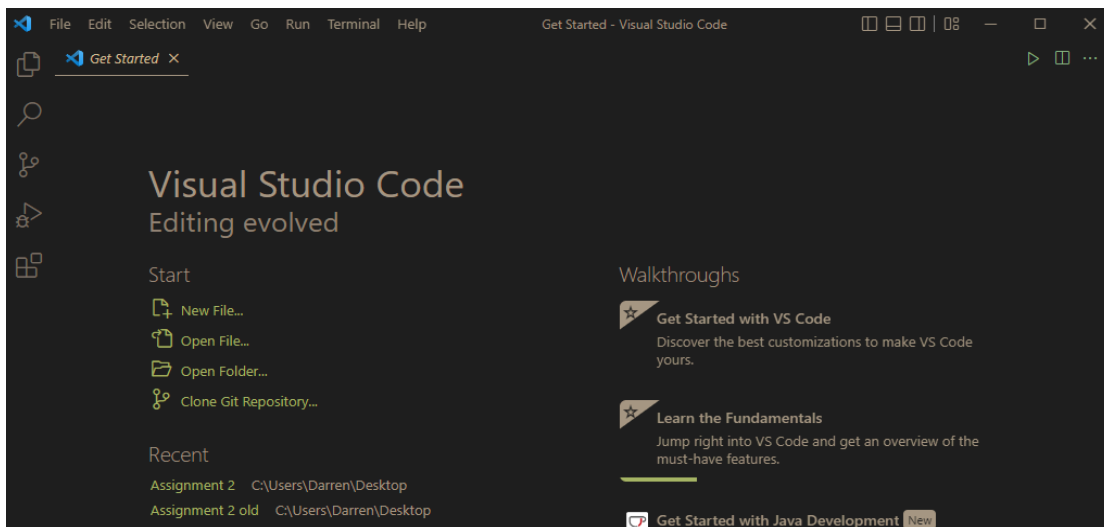
Your First Java Program

You will create a new Java file named Assignment1.java in Visual Studio Code.

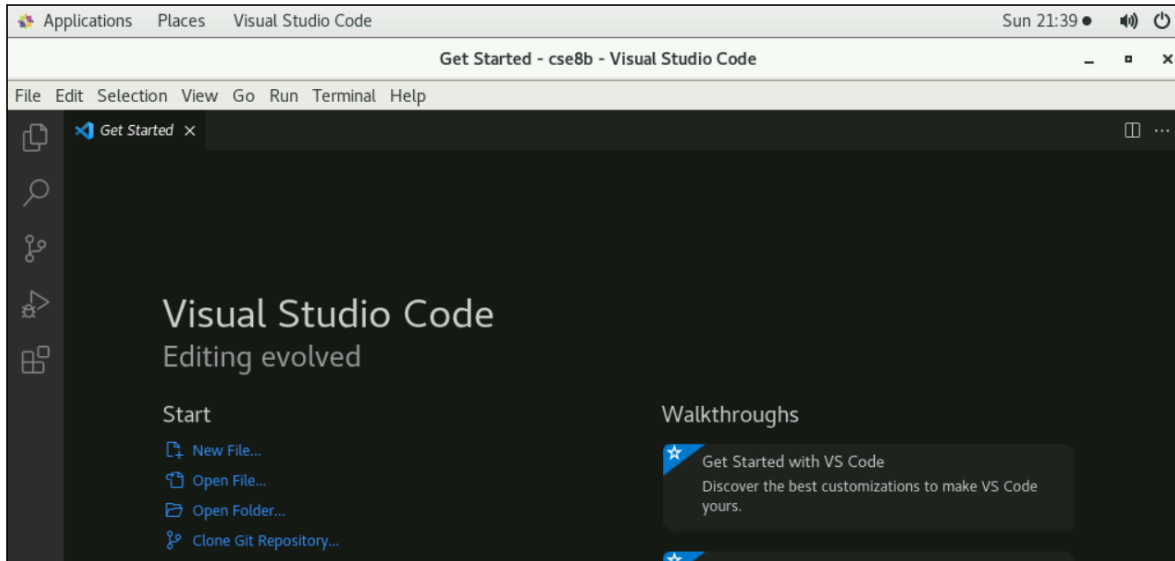
1. Open up Visual Studio Code.
 - a. If you are on the UCSD Linux Cloud or using one of the basement computers.
On the top left corner of the window, select Applications → Programming → Visual Studio Code.



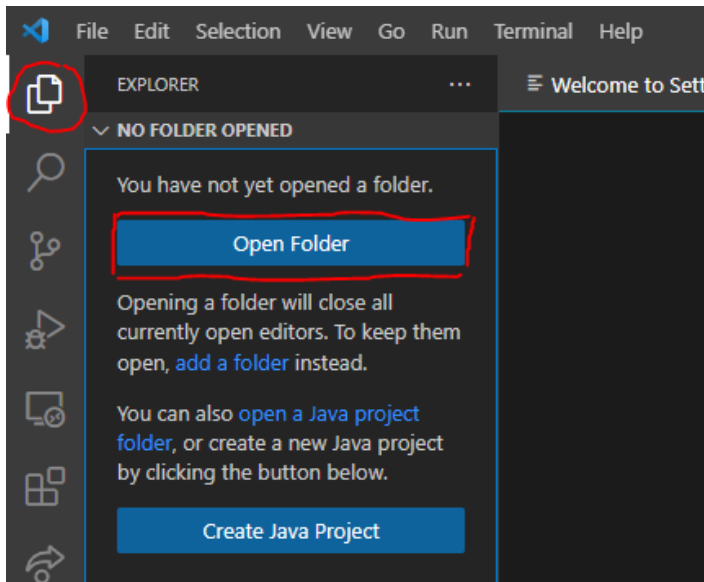
- b. If you are using a personal computer, you can open 'Visual Studio Code' like you would with a normal application.
 - c. Alternatively, if you are on the **cmd, CLI, console, or terminal**, you can type in 'code' and it should open up Visual Studio Code as well (pretty neat right?).
2. You should be prompted with a page that looks similar to the one below:
- a. Personal computer



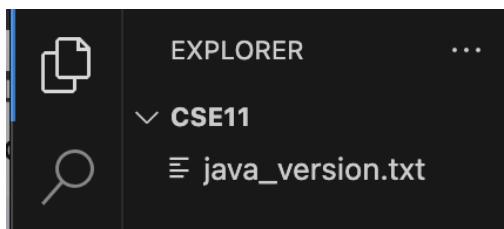
b. Linux Cloud



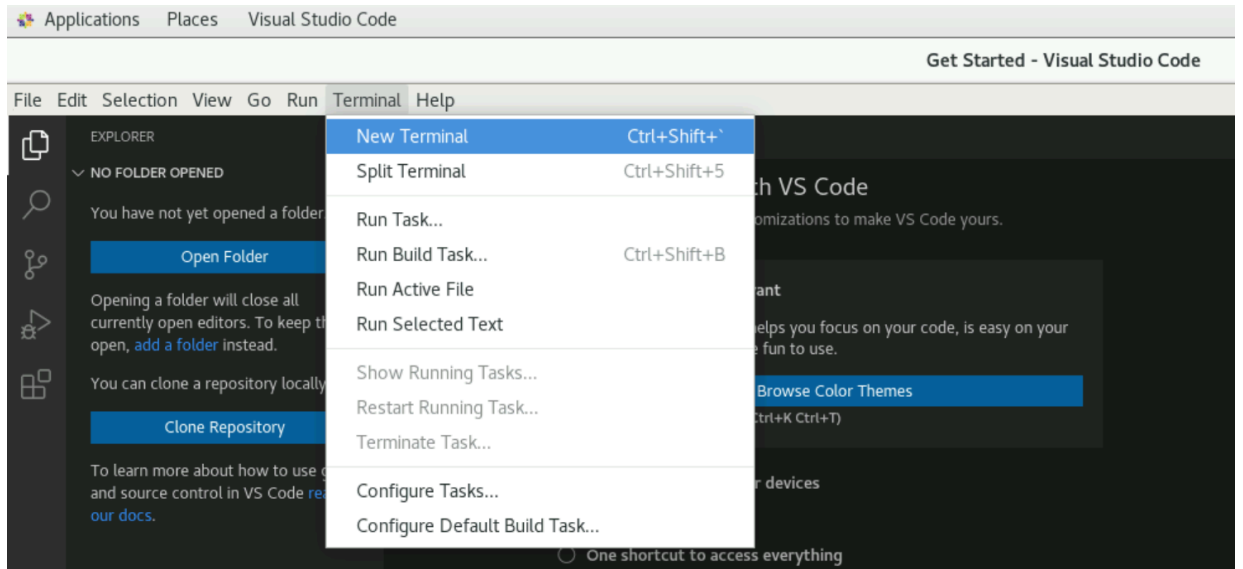
3. Let's navigate to our **cse11** directory we created before!
Click on the Explorer tab and select "Open Folder" in VS Code to locate your directory.



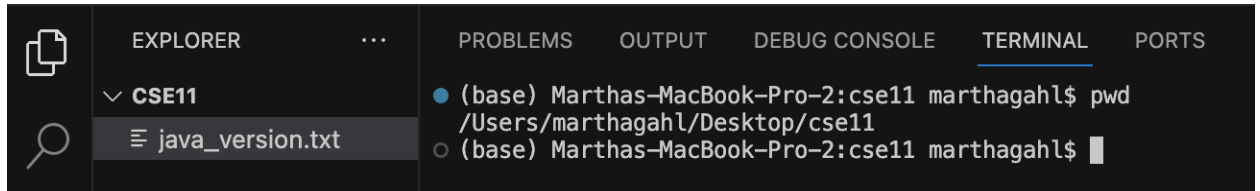
4. We should be able to see our file explorer on the left-hand side.



- Next, on the VS Code toolbar at the top, select Terminal → New Terminal.



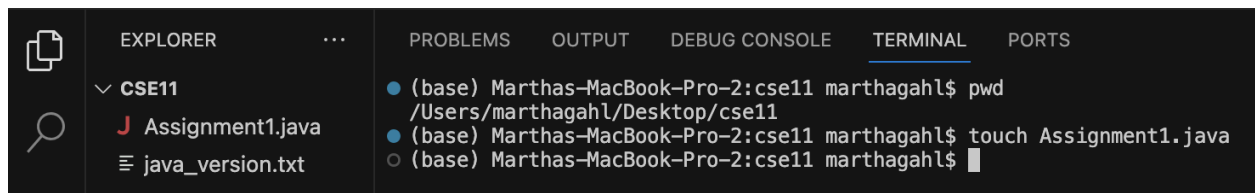
- Look familiar? You just opened up CLI in VS Code!



- Now let's create a Java file called **Assignment1.java**. You can do this either from the CLI we just opened or by simply using VS Code.

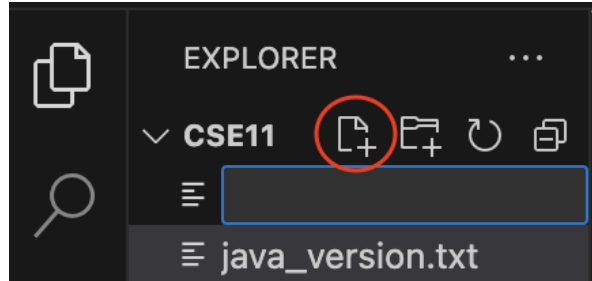
- CLI

- Create the **Assignment1.java** file using 'touch' or 'type'

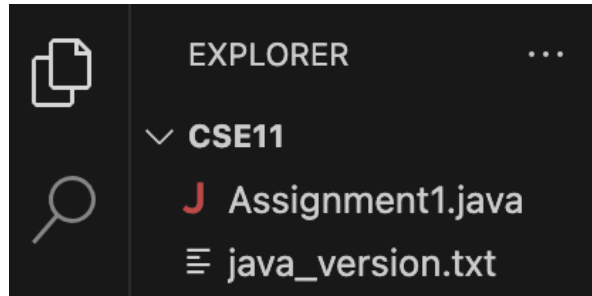


- VS Code

- Use the 'add file' icon.



ii. Name the file.



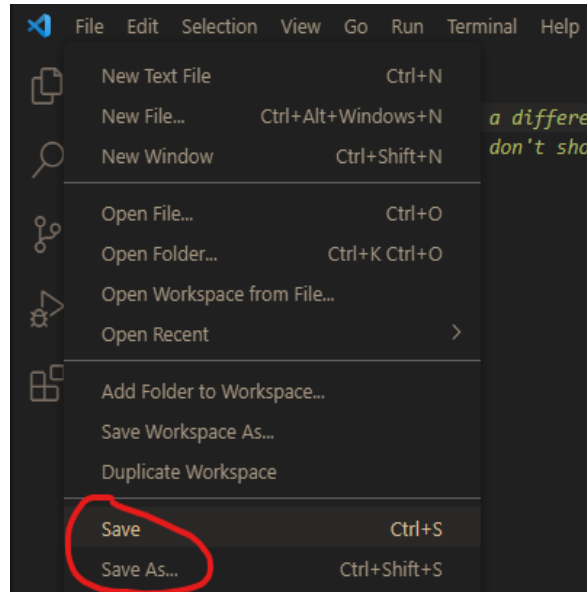
8. Type the contents of the screenshot below into your newly created Java file

```
1 public class Assignment1 {
2     public static void main(String[] args) {
3         System.out.println("Hello CSE 11!");
4
5         //Put your code below
6
7     }
8
9 }
10
```

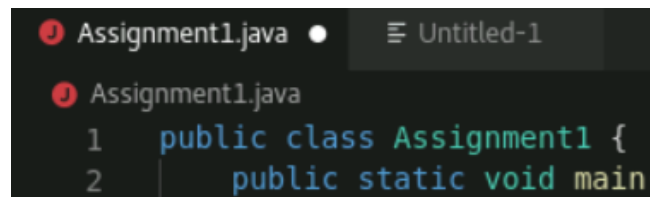
9. Then, you need to write two lines of code below //Put your code below

- a. **My name is <your name>** starting from a new line. You'll need to use the `System.out.println` method to accomplish this. Refer to the source code line that prints out "Hello CSE 11!" but replace the text between the double quotation marks with `My name is <your name>`
- b. Then on another new line, tell the class what you like: **I like <what you like>**

10. Save this file to your cse11 directory by navigating to the top left-hand corner again or by pressing `<CTRL>` and **S** (Windows) or `<CMD>` and **S** (Mac) at the same time.



You will know if your new code is saved when the white dot next to the filename at the top of the tab disappears. In the below screenshot, recent changes to Assignment1.java are unsaved.



11. After that, go back to the terminal and make sure you are within the cse11 directory. Before you run every Java code, you need to compile it first using the javac command. **Important Note:** If you make changes to a Java file, you will need to compile your code again before you can run the file with the updated changes.

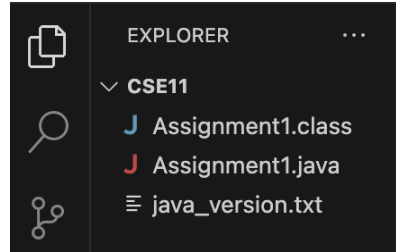
- Linux and macOS:

```
$ javac Assignment1.java
```

- Windows:

```
> javac Assignment1.java
```

Notice that a brand new .class file appeared in the cse11 directory after compiling! (If it doesn't show up in VS Code, you can still see it when you call "ls" or "dir" in the CLI.



Then you can run the program using the java command.

- Linux and macOS:

```
$ java Assignment1
```

- Windows:

```
> java Assignment1
```

You should see three lines of output in the terminal.

```
Hello CSE 11!  
My name is Martha  
I like ice cream
```


Submission Instructions

VERY IMPORTANT: Please follow the instructions below carefully and make the exact submission format.

1. Go to Gradescope and click on PA1.
2. Click the DRAG & DROP section and directly select the two required files (**java_version.txt** and **Assignment1.java**). Drag & drop is fine. *Please make sure you don't submit a zip. Just the two files solely. Make sure the names of the files are correct.*
3. **You can resubmit unlimited times before the due date.** Your score will depend on your final submission, even if your former submissions have a higher score.
4. The autograder is for grading your uploaded files automatically. For this first assignment, the autograder (grading script) will display to you the result/score of your submission immediately.
5. Your submission should look like the below screenshot (order of file doesn't matter). If you have any questions, feel free to post them on Piazza!

Submit Programming Assignment

? Upload all files for your submission

SUBMISSION METHOD

Upload GitHub Bitbucket

Add files via Drag & Drop or [Browse Files](#).

NAME	SIZE	PROGRESS x
java_version.txt	9 b	<div style="width: 100%;"></div>
Assignment1.java	0.2 KB	<div style="width: 100%;"></div>

STUDENT NAME (OPTIONAL)

Enter student name **v**

Upload **Cancel**