# AWS Cloud9: A Linux System in Your Browser

As explained in the previous reading about setting up a Linux-compatible environment, there is a service provided by Amazon called AWS Cloud9 which allows you to access a Linux terminal and integrated development environment (IDE) entirely through your web browser. There are several ways that you can access this service for free. We will show you how to do this as a way of ensuring that you have access to the proper tools regardless of what kind of computer system you have.

There are several different types of AWS accounts because AWS is run as both a professional service and an educational service. Here we'll briefly explain the differences and give some tips about signing up. You can also view Amazon's own support page that discusses this same topic: <https://docs.aws.amazon.com/cloud9/latest/user-guide/setup-student.html>

## Option 0: Not using AWS Cloud9

If you prefer not to use AWS Cloud9, then you will need to install a Linux-compatible C++ coding environment and compiler on your own computer as described in the previous reading.

## Option 1: Signing up for a normal AWS account with the free tier

This method of signing up for AWS gives you a normal AWS account such as a professional would use to access cloud services provided by Amazon. The free tier of service, which is available for 12 months to new users, gives you access to all that you need to accomplish the homework assignments in this course. (If you already have an AWS account for other purposes, then there is a chance that your usage for this course would be in excess of the free limits and incur some small charges to your AWS account. We estimate that running the Cloud9 IDE even up to 20 hours a week would only cost about $1 USD per month.)

**Pros of creating a normal AWS account:**

* You will create a normal AWS account that has no special limitations. (If you already have your own AWS account, you can simply use that.)
* Your account is eligible for the bonus starter credits that Amazon has allowed us to provide to you.

**Cons of creating a normal AWS account:**

* You must leave payment information on file with Amazon, such as credit card number and phone number. (However, it is likely that you will not incur any charges.)

If you would like to create a normal AWS account, visit the following link and click "Create an AWS Account." <https://aws.amazon.com/console/>

After logging successfully to the AWS Console, you're ready to move on to the next reading tutorial for this week, where you will create a Cloud9 workspace.

## Option 2: Signing up for a free AWS Educate Starter account

This type of AWS account has special limitations compared to a normal AWS account. It is still useful for accomplishing the assignments in this course. You can see a table with the specific differences between normal AWS and Educate Starter accounts here: <https://www.awseducate.com/faqs?app=2#fa0Po000000C0dlSEAR>

We primarily recommend this option for those students who do not have access to a credit card, or for those who do not want to give any payment information to Amazon. However, you must already have a school email account or be able to prove enrollment through other means, as mentioned here: <https://www.awseducate.com/faqs?app=2#fa0P1N00000MPeYLUA1>

**Pros of creating an AWS Educate Starter account:**

* You do not need to enter any payment information.

**Cons of creating an AWS Educate Starter account:**

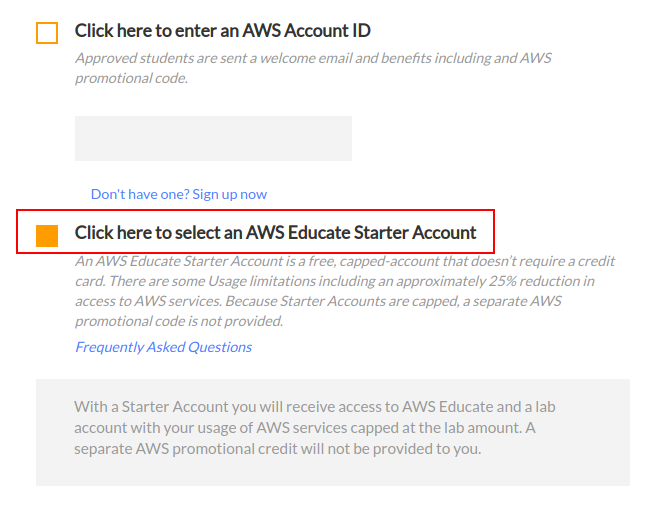
* For this account type, you must have a .edu email address already. If you don't have one, you'll need to provide proof of enrollment through other means.
* Your account cannot be converted into a professional AWS account later on. You cannot continue to use the account after you graduate.
* Your account is not eligible for the bonus starter credits that Amazon has allowed us to provide to you.
* The Starter account has a 1-hour timeout during usage, which may require you to log in again every hour. To prevent losing any work, we will show you how to enable auto-saving in the Cloud9 editor.
* You cannot be logged in to a normal AWS account and a Starter account at the same time. Please only use one type of account or the other.

If you would like to create an AWS Educate Starter account, follow this link and click "Join AWS Educate": <https://aws.amazon.com/education/awseducate/>

(Note that you must also use this link to sign back into your AWS Educate Starter account, not the normal AWS Console link. If you try to log in through the normal AWS link, it will not recognize your account.)

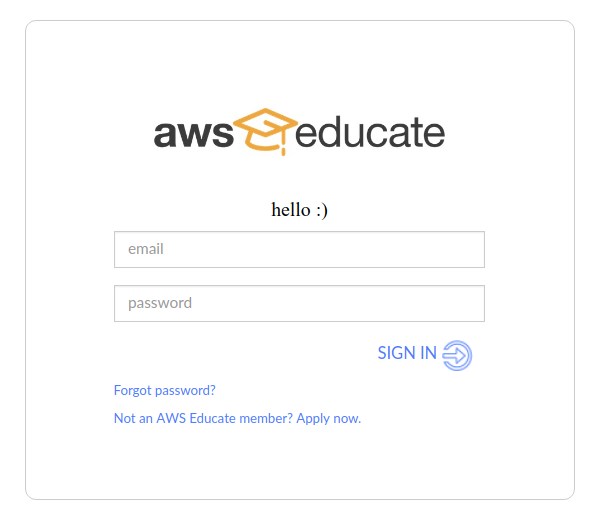
**There are some special things to note about signing up for and logging into a Starter account:**

During the signup process, you must be sure to select the AWS Educate Starter Account type:

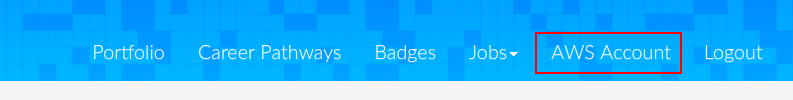


Remember that to log back into your Starter account, you need to use this special login link, not the "normal" AWS login link: <https://aws.amazon.com/education/awseducate/>

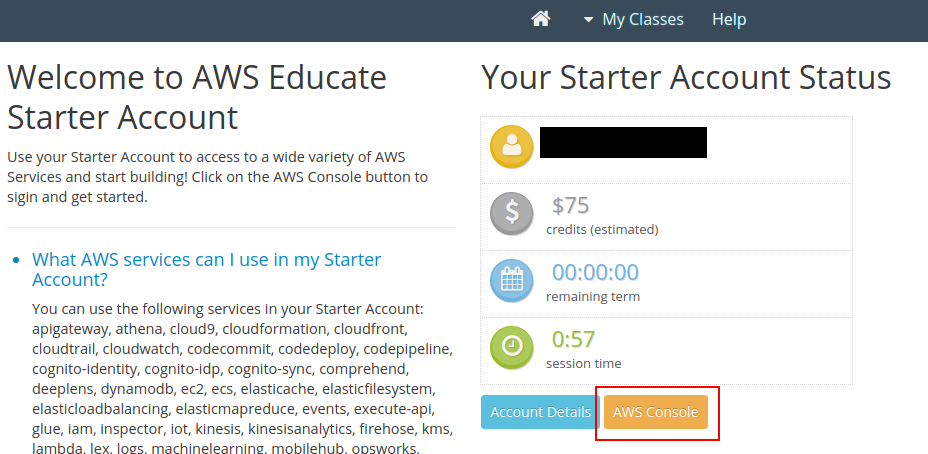
The special login screen looks like this:



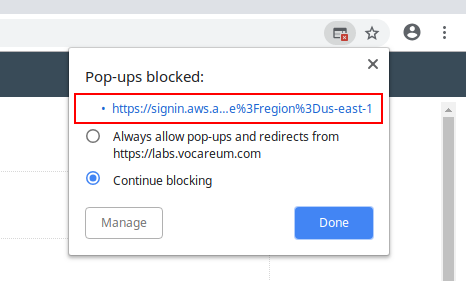
After logging in to your Starter account, the first screen will show this menu, where you must click "AWS Account" to proceed:



Then you will see another welcome screen like the one below, where you must click the "AWS Console" button to proceed:



**This might cause a "pop up" window to appear, which your browser might block.** If you do not see the AWS Console site appear, then check your browser's address bar to see if a popup was blocked. In Chrome, you can choose to override the block by clicking the blocked link:



Finally, you will arrive at the AWS Console, and you can continue with the next reading tutorial here on Coursera, where we will show you how to create a Cloud9 environment.

AWS Setup II: Creating a Cloud9 Environment on AWS

Now that you've created an AWS account, we can guide you through creating your first Cloud9 environment.

A Note About Professional Security on AWS

If you are using a limited AWS Educate Starter account, then this warning does not really apply to you, but you may want to read it anyway to understand more about AWS.

Although AWS Cloud9 gives you access to a real computer system running in the cloud, please remember that in general, AWS is a professional tool makes you the **virtual network administrator** of various cloud services offered by Amazon. You also have the power to manage additional virtual "users" that you can authorize to access your services. If you are using a normal AWS account instead of an AWS Educate Starter account, then as you follow the instructions below, you'll see a warning that you are using Cloud9 as the AWS root user. This means that you are logged into Cloud9 as a user who has **full control** over your entire AWS account. The root user can add or remove Amazon services and also manage other virtual users within your personal AWS network. In theory, you could execute certain malicious code or type certain commands within your Cloud9 environment that would affect other services on your AWS account and incur charges or disrupt your other AWS service operations.

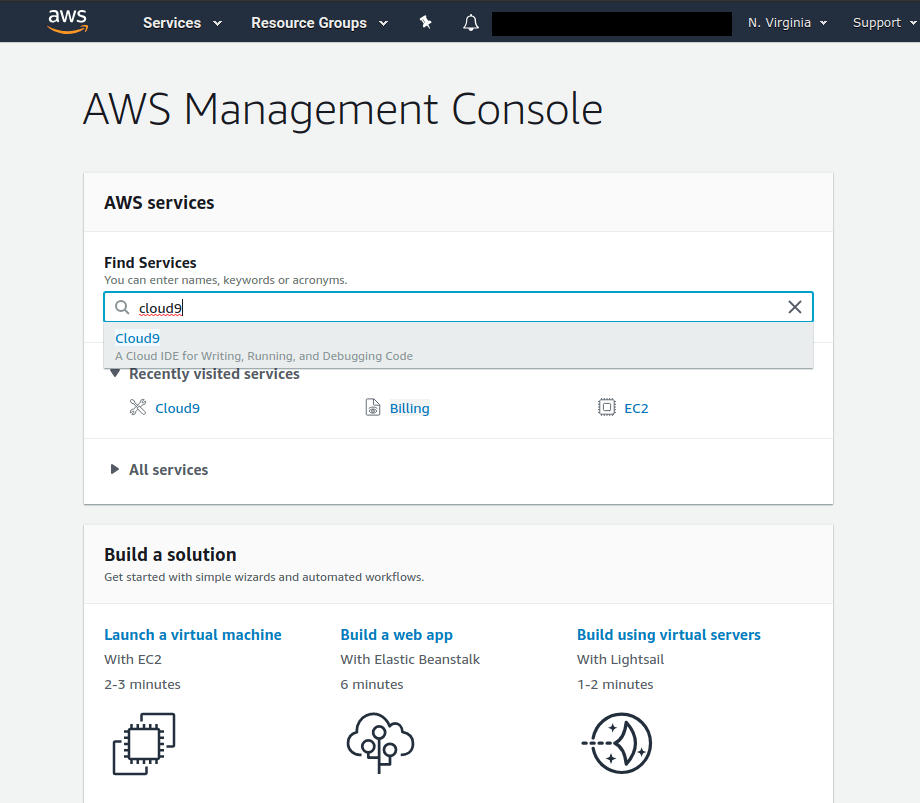
So, if you use your AWS account for other purposes apart from this course, then you should follow the best security practices outlined at the following link: <https://docs.aws.amazon.com/cloud9/latest/user-guide/setup.html> This will guide you through creating a Cloud9 user group and a limited Cloud9 user, which has only enough permission to access the Cloud9 environment and work on the assignments. If you do so, then you will need to grant your limited Cloud9 user account its own username and password, and the user creation setup will give you a special link that you must use to log into AWS with your new username.

However, if you are new to AWS and you **only** intend to work through the assignments in this course, you may be able to simply get started without taking those additional steps. Please be careful to only execute the commands in your Cloud9 terminal that we suggest to you. **Do not attempt to "share" access to your Cloud9 workspace with anyone.**

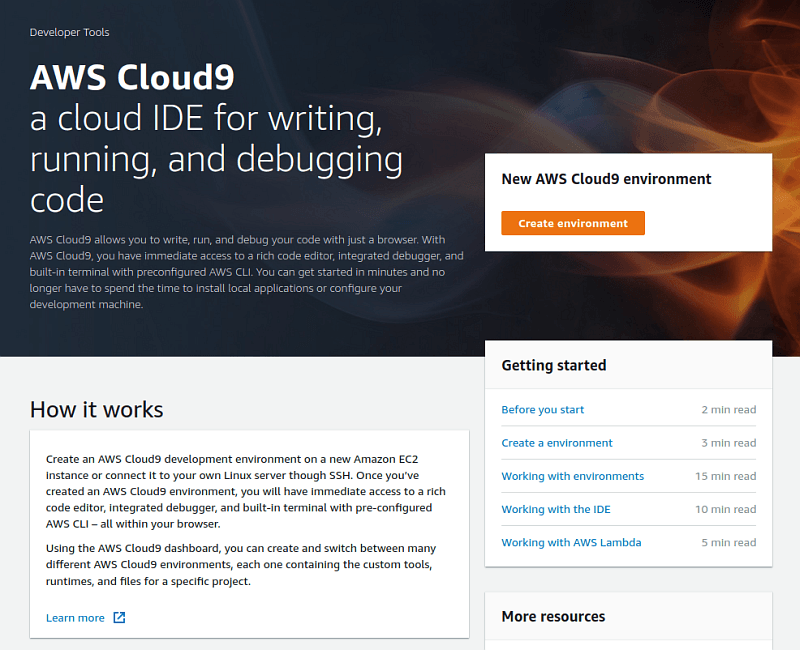
Creating Your First Cloud9 Workspace

After logging into your AWS account, you will see the AWS Management Console screen, where you can find all the services that AWS offers.

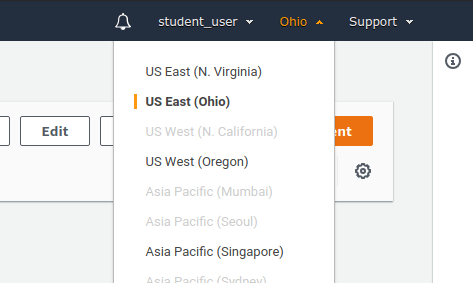
Now we need to find the Cloud9 service. You can search for it by typing "cloud9" in the search box and clicking on the Cloud9 option that appears:



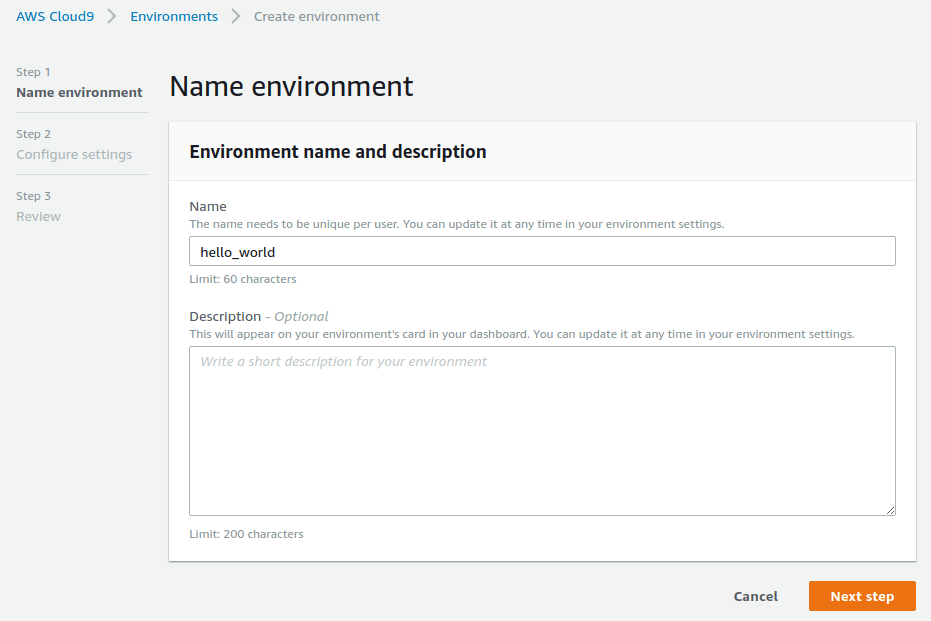
If you have never used Cloud9 before, then you will see the following welcome screen:



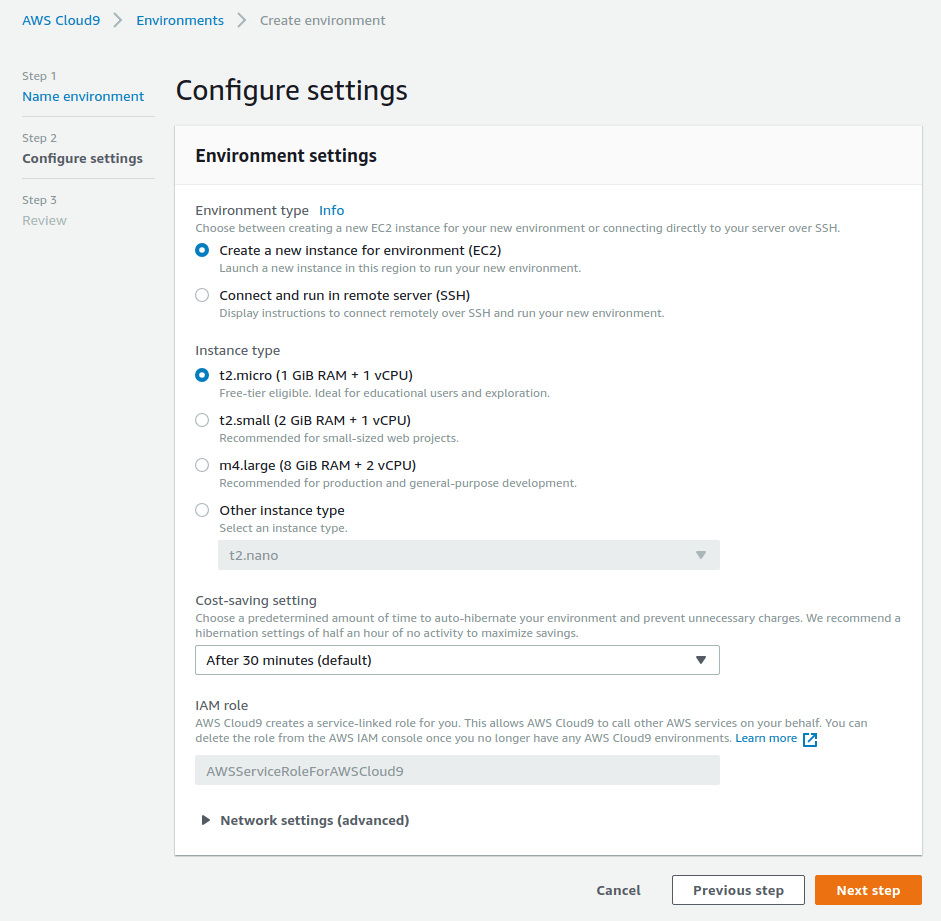
Before we create the environment, it's important to choose your AWS region. **You will always want to select the same region when you use AWS, or your existing workspaces won't appear!** So, choose a region close to you, and make sure to always select the same one from now on when you log into AWS.



Now, you can click "Create environment" to continue. You will then be able to enter a name for your environment workspace. For example, "hello\_world":

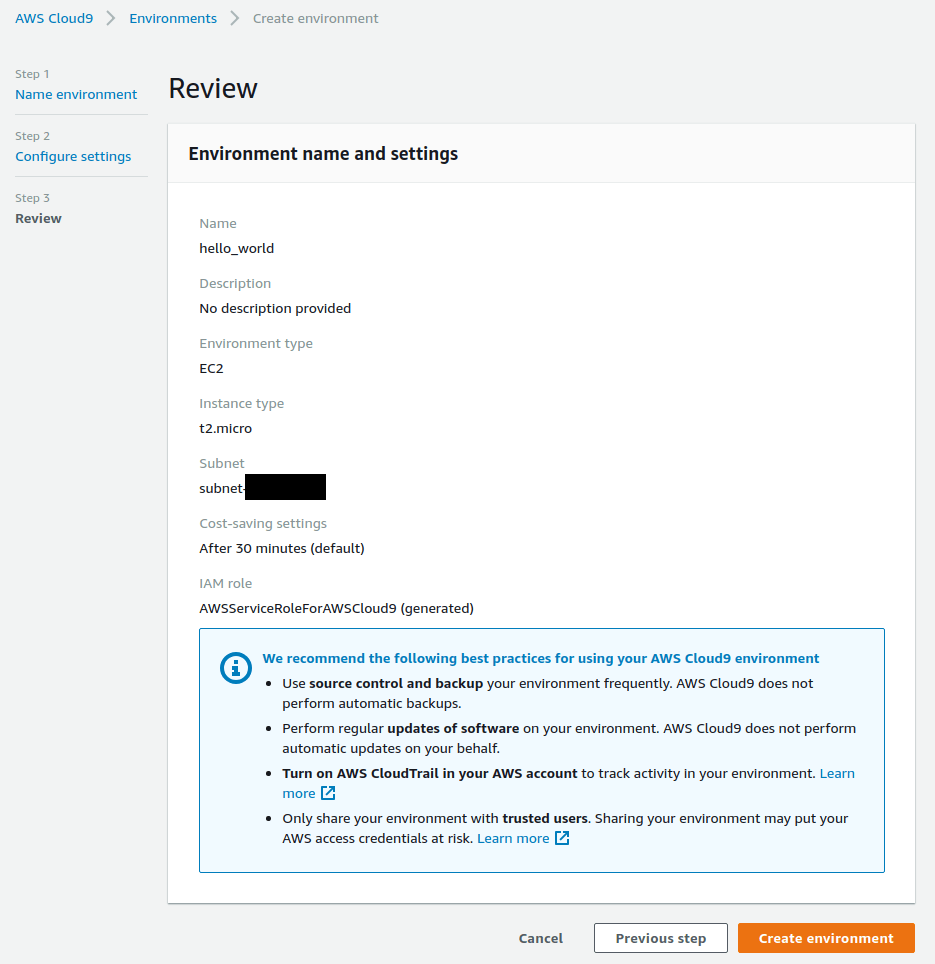


On the Configure Settings screen, you can leave the default settings and continue. Note that your workspace is set up to automatically go to sleep after 30 minutes of inactivity, which helps to prevent wasting valuable time on your AWS account:

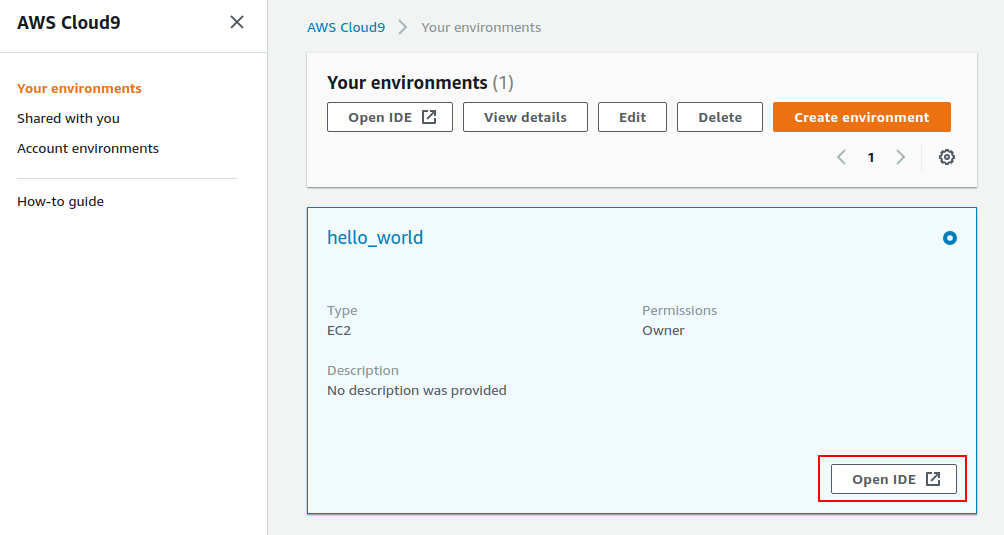


If you are offered a choice between "Amazon Linux" and another Linux distribution (variant) here, then please note that currently, our instructions assume that you have chosen Amazon Linux.

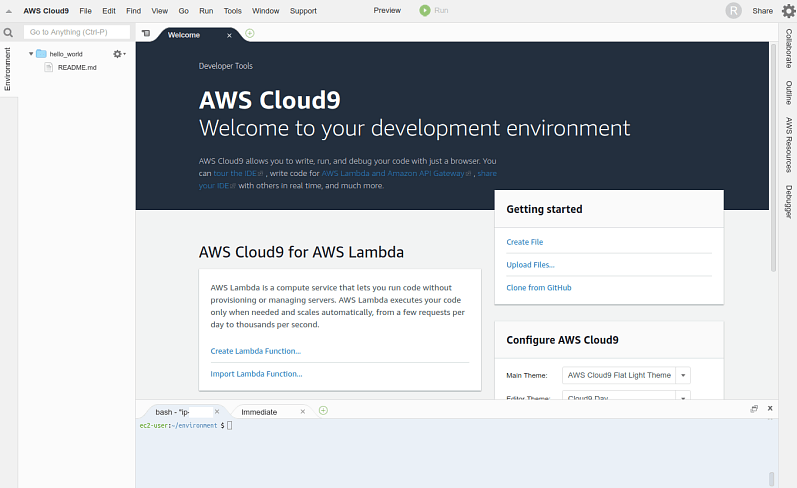
After that, you will see this confirmation screen, and you can click "Create environment":



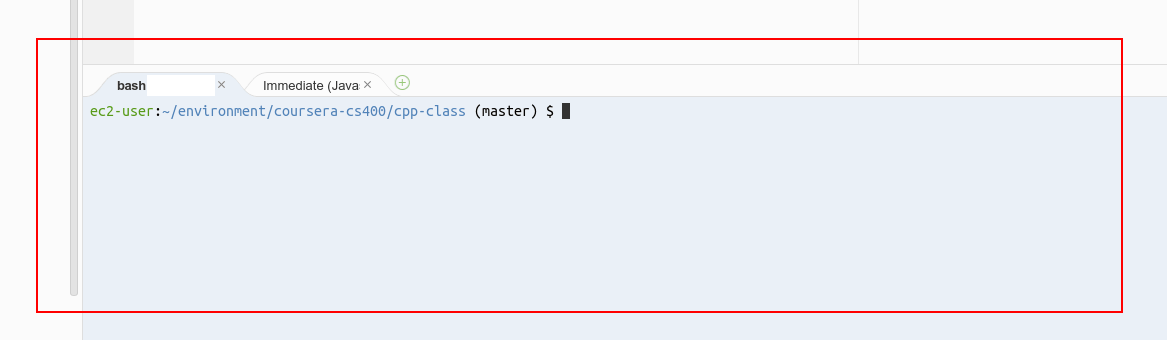
Now, you may see a brief loading screen as your environment boots up. Or, you may see this dashboard screen where you can choose to launch the environment by clicking "Open IDE":



After the environment finishes loading, you'll see the main IDE workspace screen:



**[Update: Tuesday, Feb 19, 2019] Now, we need to update your compiler to a more recent version.** Look at the terminal command prompt at the bottom of the environment. It's the part highlighted in red here, although yours might look a little different:

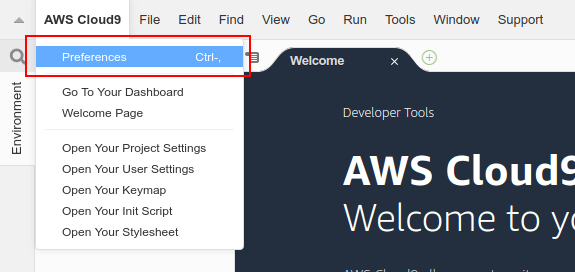


You need to type the following text command into the prompt after the **$** sign and press Enter. Be sure to type the entire command, which begins with "sudo" and ends with "gcc72-c++". You might want to copy and paste the command exactly:

sudo yum install -y gcc72 gcc72-c++

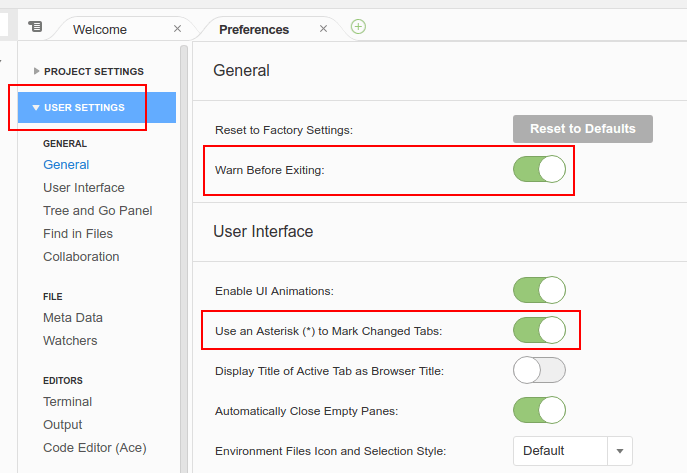
This command invokes the automatic installation and update system on this version of Linux, and it updates the compiler to a specific version. You don't need to know exactly how this command works, but here's something important to point out. **"sudo"** means the command will run with system administrator permissions, which is potentially dangerous if you don't know what the command actually does. In general, you should be very cautious about entering any terminal commands, but *especially* if the command uses "sudo". **For this course, please only trust terminal commands that we explicitly tell you about.** Community help sites like StackOverflow may show terminal commands, but even though such sites use community curation to prevent misleading or false information, you should still be careful about any commands you enter on your system.

Next, let's set up some autosave features in the text editor to prevent you losing any work due to network connectivity issues. At the top left, click "AWS Cloud9" and then "Preferences":

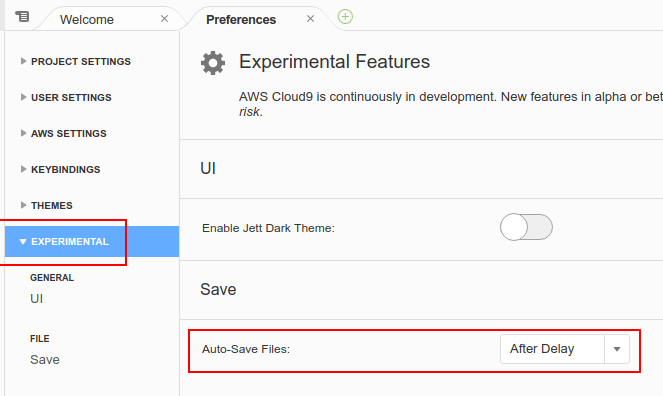


In the Preferences, under User Settings, we recommend selecting these options:

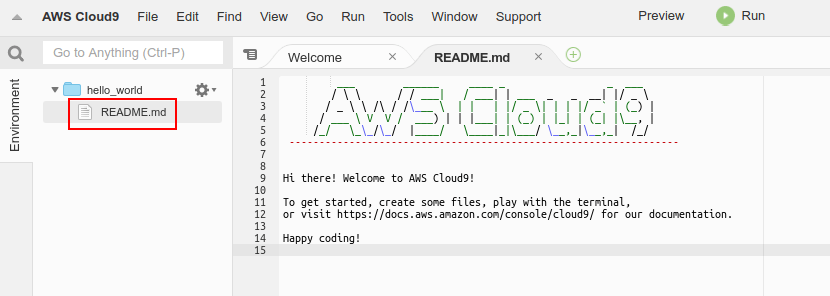
* Warn Before Exiting: This will display a warning popup in your browser whenever you try to close the main Cloud9 workspace window or tab.
* Use an Asterisk (\*) to Mark Changed Tabs: Whenever you have unsaved changes in an open text file, this will display a "\*" mark in the open file tab.



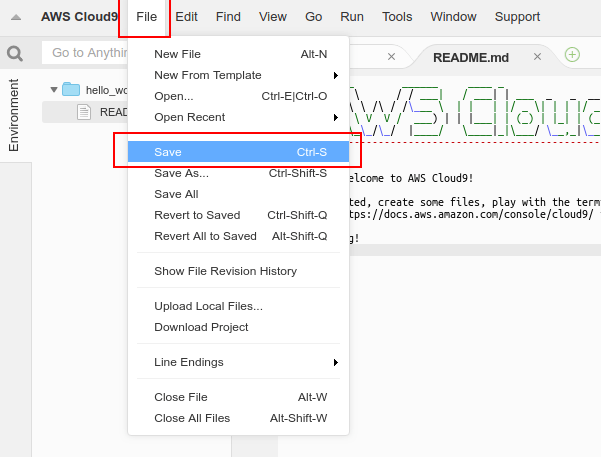
Also, under "Experimental" preferences, you can set "Auto-Save Files" to "After Delay". This will automatically save your changes after a few seconds.



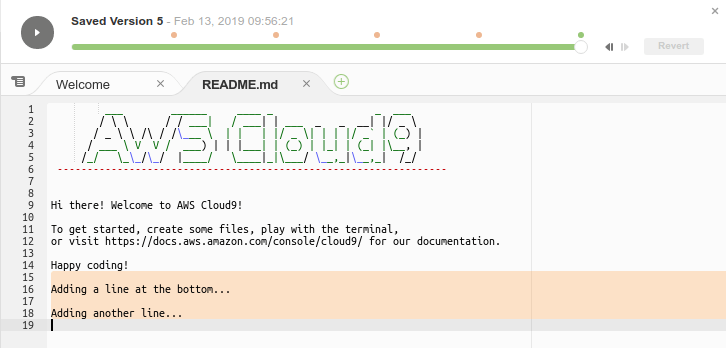
Now you can double-click the README.md file in your file list on the left, and it will open as an editable file in the right pane. Try making some edits to the file. You will see a "\*" briefly appear in the filename tab, and then the "\*" will disappear when your changes are automatically saved.



Remember that you can also manually save a file by clicking File > Save in the AWS menu bar (not in the browser menu bar).



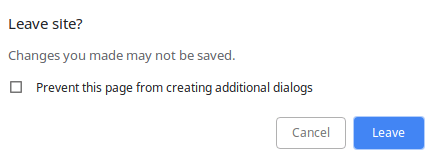
But since your changes will be saved automatically very frequently, you may want to see the history of changes. If you select File > Show File Revision History, you will see a screen like this one, where you can play back the history of edits to your file, and even undo the changes by clicking Revert:



This method of version control is not as advanced as the terminal program Git, but it is very easy to use. When you are done viewing the edit history, click File > Show File Revision History again to disable the history view and continue editing the file.

**(Important note for advanced git users:** If you intend to use Git to manage your own file history, then you should probably be careful about the Cloud9 file history feature, and perhaps use git alone for this purpose. We are not sure how compatible the Cloud9 revision history is with git's own version tracking.)

If you are done experimenting with the workspace for now, you may want to close the Cloud9 tab in your browser. When you try to close the browser tab, you will see the warning message we enabled in preferences, alerting you that you may have unsaved work. Here is what it looks like in Chrome:



If you're sure you want to quit, just click the Leave button and the browser tab will close. The cloud server that runs your workspace will automatically go to sleep after 30 minutes to prevent waste on your AWS account.

Now, you're ready to continue to the next lesson!

# Updating the C++ Compiler

On Cloud9, if running our makefiles with "make" gives an error message about "--std=c++14", you probably forgot to run this command that was mentioned in a previous tutorial:

**sudo yum install -y gcc72 gcc72-c++**

Please run this command to update the version of the compiler on your Cloud9 workspace. This assumes you had chosen "Amazon Linux" as your Cloud9 operating system

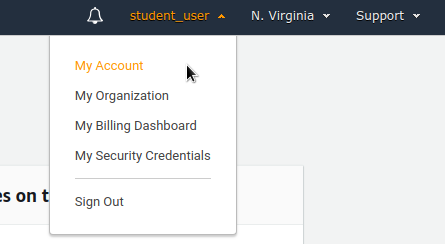
## Bonus: Redeem Extra AWS Starter Credit

When you click on the "Open Tool" button below, you will be forwarded to a special page on Amazon's AWS Educate site, where you can redeem an additional amount of AWS promotional credit. You can always get started for free with Cloud9 on a normal AWS account, but this gives you bonus credit that you can use to do more with your AWS account, and the value will carry over if you choose to use your AWS account beyond this course.

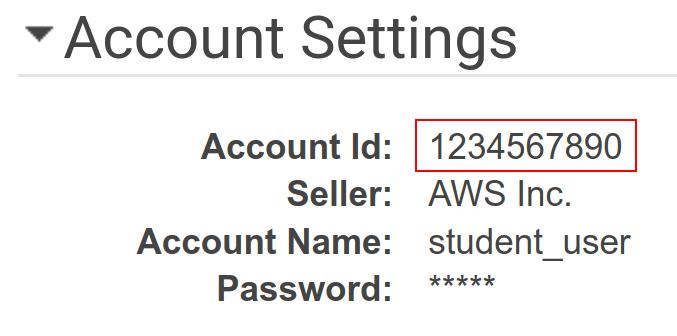
**This is not available for AWS Educate Starter accounts.**The promotion will only work for a normal AWS account, as described in the previous reading.

**Also note:** If you had previously tried to create both a normal AWS account and an AWS Educate Starter account with the same email address, following this procedure will make your AWS Educate Starter account inaccessible. You will only be able to log in to your normal AWS account thereafter.

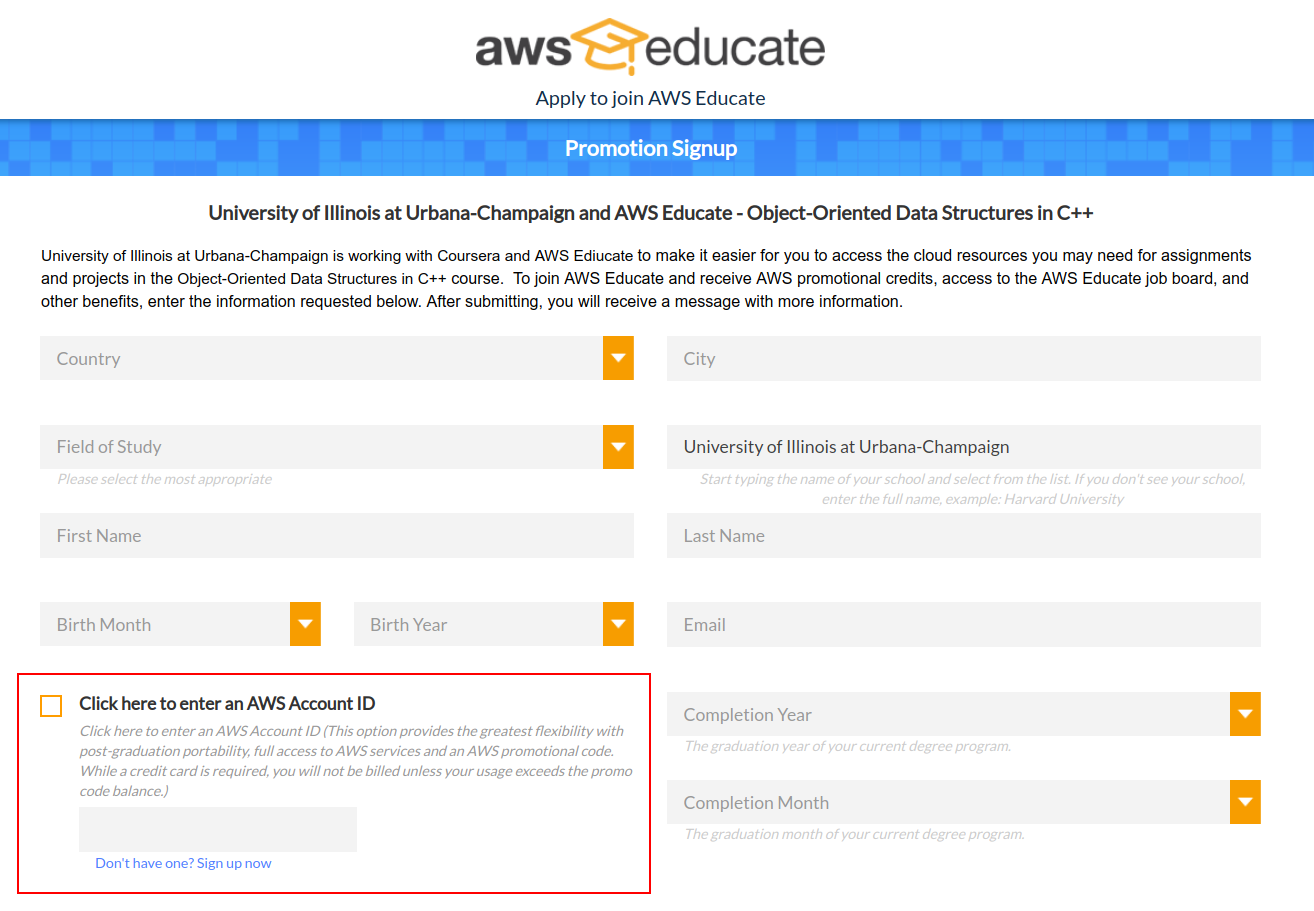
Before you can redeem the code, you will need to find your AWS account ID from your normal AWS account. You can find it by clicking your user name in the upper-left of the AWS console page and selecting My Account:



After that, you'll see the account number that you need:

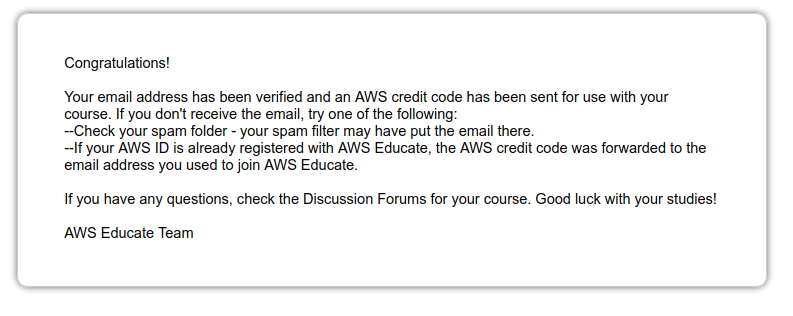


Then sign out of AWS. You can proceed to click on this LTI Item's button below, which will take you to this signup screen:



Note the box marked in red above. You must check the box and enter your AWS account ID number while you sign up. (The page incorrectly describes this as optional, but it is required.)

When you are done signing up for the promotion, you should see this message appear:



Then watch your email inbox for any further information about the promotion credit. You may be emailed a promo code to enter on AWS manually. Or, you may be awarded the credit directly.

**Passed** 100%

This course uses a third-party tool, Bonus: Redeem Extra AWS Starter Credit, to enhance your learning experience. The tool will reference basic information like your name, email, and Coursera ID.

<https://www.awseducate.com/PromotionSignup?pcode=TPIAR4>

## Recommended: Cloning the GitHub Repository Using the ****git**** Terminal Command

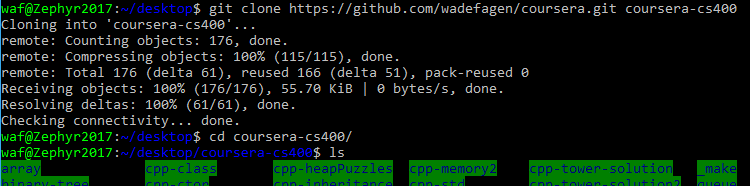
After following the previous readings, you may already have git installed on your command line! Open you Linux command line (this is the **Ubuntu** app if you're on Windows using WSL) and navigate to your desktop (**cd ~/desktop** if you're using Windows WSL and followed the setup hints in the earlier reading lesson).

**Note:** If you are using AWS Cloud9, then instead of using a "desktop" directory you will probably want to begin in the **~/environment** directory, which corresponds to the file list on the left side of the Cloud9 workspace. You can type **cd ~/environment** in the Cloud9 terminal to go there.

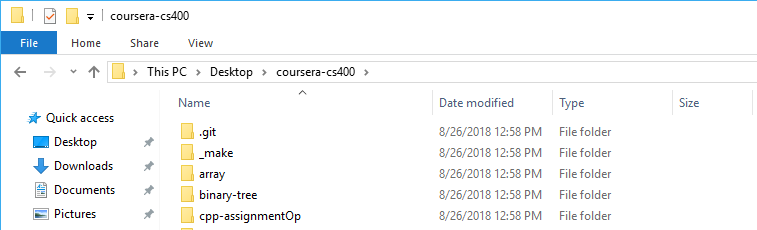
Once you are in the ~/desktop directory (or in the ~/environment directory), you will clone the GitHub repository for this course with the following command:

git clone https://github.com/wadefagen/coursera.git coursera-cs400

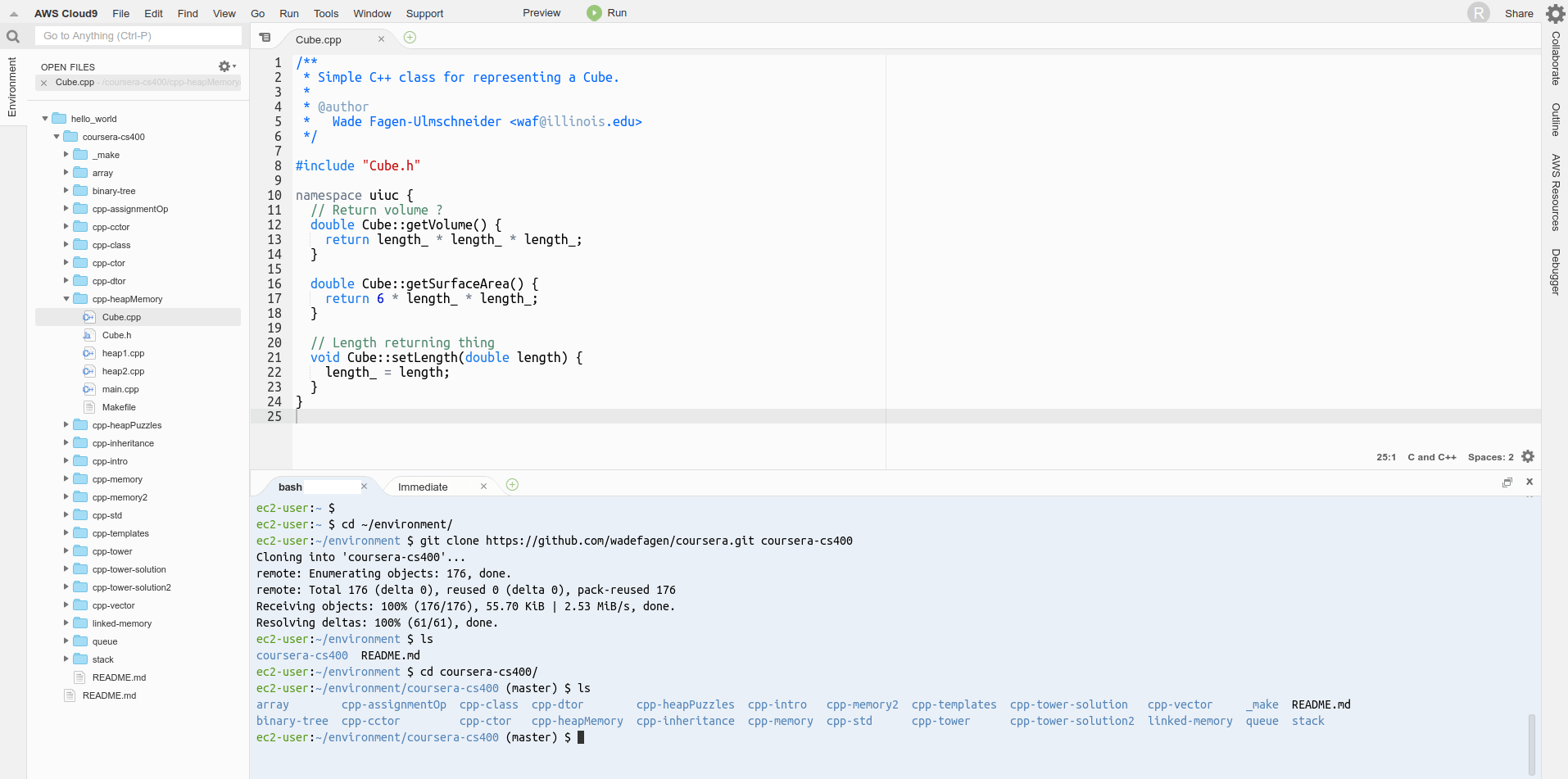
This command will create a **coursera-cs400** directory that contains the source code files for this semester. Here's a screenshot showing these steps running on my computer:



You can also view these files on your Windows Desktop:



And on Cloud9, this is what it would look like. We typed the necessary commands in the terminal to download, and then we browsed to one of the files in the left-hand file list to open it.



Now you can explore the file contents and view the code examples shown in lecture.

## Keeping the Local Repository Updated

If you have cloned the git repo (repository) locally using the "git" command, then you can also receive updates that the course staff might make. For example, we may add small bug fixes or improvements, as well as further examples. Here's how you can make sure you have the latest version.

First, if you have made changes to the example files that you want to keep, you should manually make a backup copy of these to another directory. (There are advanced git commands that can do something like this for you, but for now, it's easiest to make a backup copy of your work manually.)

Then, in the bash terminal, make sure you are in the coursera-cs400 directory where you have cloned the repo locally. The terminal commands below should be entered in the bash terminal one at a time. These commands will ensure you are on the main "master" branch, reset any changes you have made, and then pull the latest versions from the server:

git checkout master

git stash

git pull

After you do this, you should be sure to do "make clean" before running "make" again in any of the example directories. That way, you will force the example to be rebuilt completely using the latest version of the code.

If you experience any confusing git error messages while trying to update your local repo, the simplest solution to clone a fresh copy of the entire repo as described in the earlier steps above, being sure to clone to a new directory locally. The newly cloned copy will have the latest version.