SIT 725 Prac 1
Basics of Programming and GIT

Contents

Basics of Git

Git is software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows.

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is easy to learn and has a tiny footprint with lightning fast performance. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like cheap local branching, convenient staging areas, and multiple workflows.

Basics of Git Cont ...

Homebrew is a package manager for macOS (and Linux, too). It is one of the first tools you'll need to set up a local development environment for programming on a Mac. Use it to install (and remove) software programs for the terminal, or command line.

Installation on Mac :

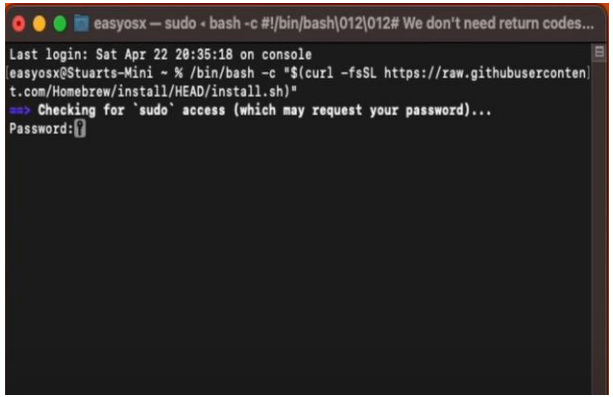
- Open Terminal
- Install HomeBrew using the command :
- /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
- brew install git

Installation on Linux(Ubuntu) :

- Open Terminal
- sudo apt-get update
- sudo apt-get install git-all

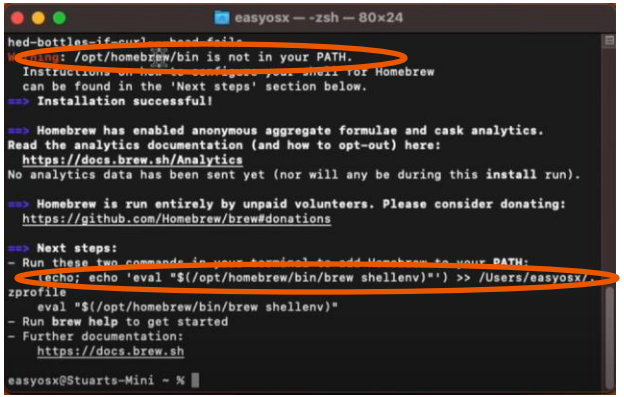
YouTube video :
installation on MacOS
https://www.youtube.com/watch?v=IWJKRmFLn-g
Installation on Ubuntu
https://www.youtube.com/watch?v=pRsHimqix24

Installation –macOS steps (video start from 01:32-03:45)



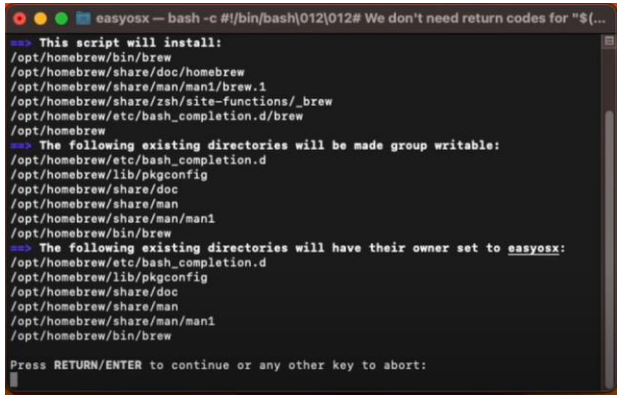
Step 1

1. Copy the code paste into Terminal
2. Type in your admin password when prompted



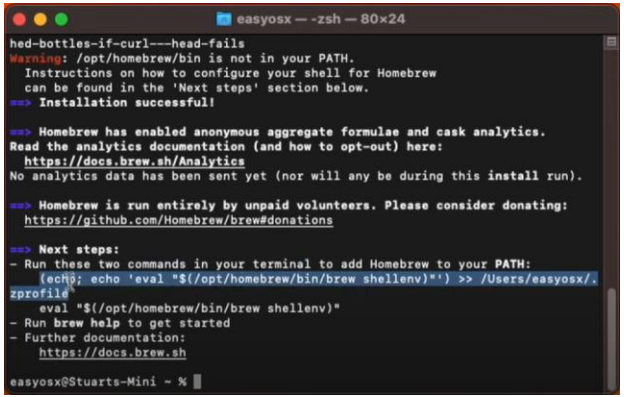
Step 3

1. Once it's done, you have Brew installed, but you're not done yet. (Brew is not in your path)
2. Set the path where HomeBrew stores its packages and where you will be modify it.
3. Look at "Next steps"



Step 2

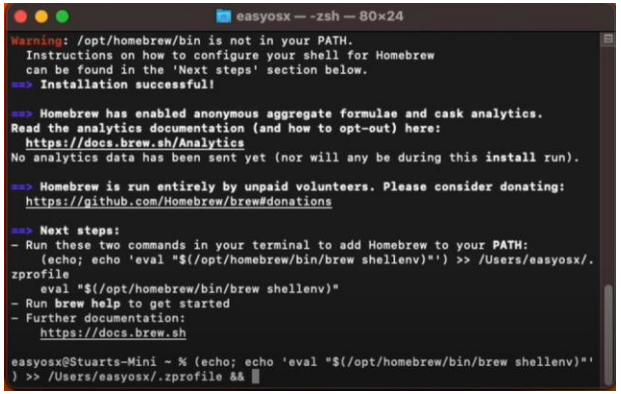
1. You will be presented with a list of what is going to be downloaded and installed and from where.
2. Hit Enter
3. Use 1-2 mins to download and install



Step 4

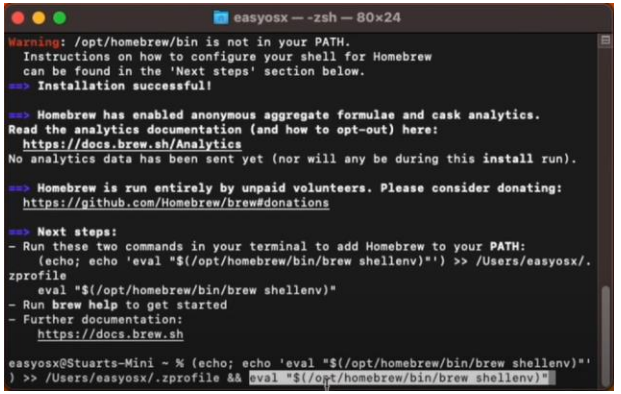
1. Highlight and copy the first command, paste it into the next line.

Installation –macOS steps Cont



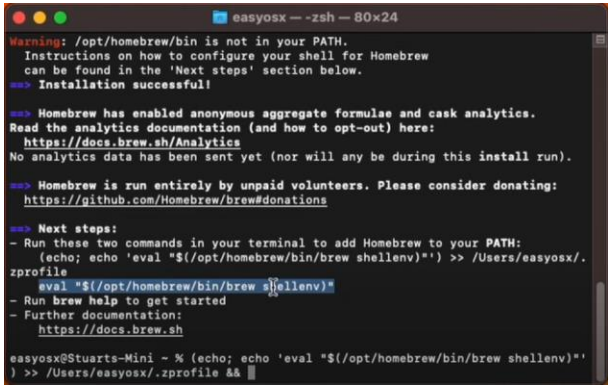
Step 5

1. Add two ampersands and an additional space after the pasted code.(This will tell the Terminal to run each command in order rather than you having to run each command separately.)



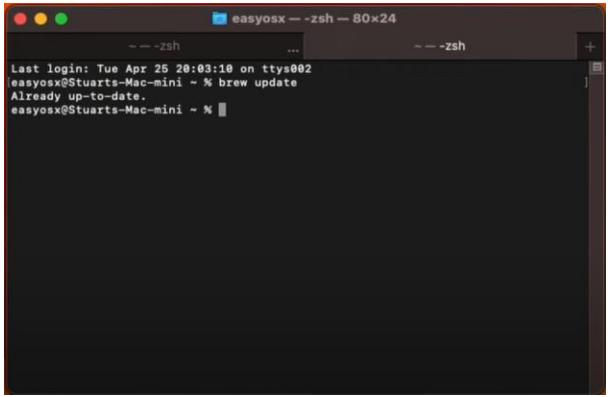
Step 7

1. Hit Enter
2. Successfully added HomeBrew to our path and have it ready to go



Step6

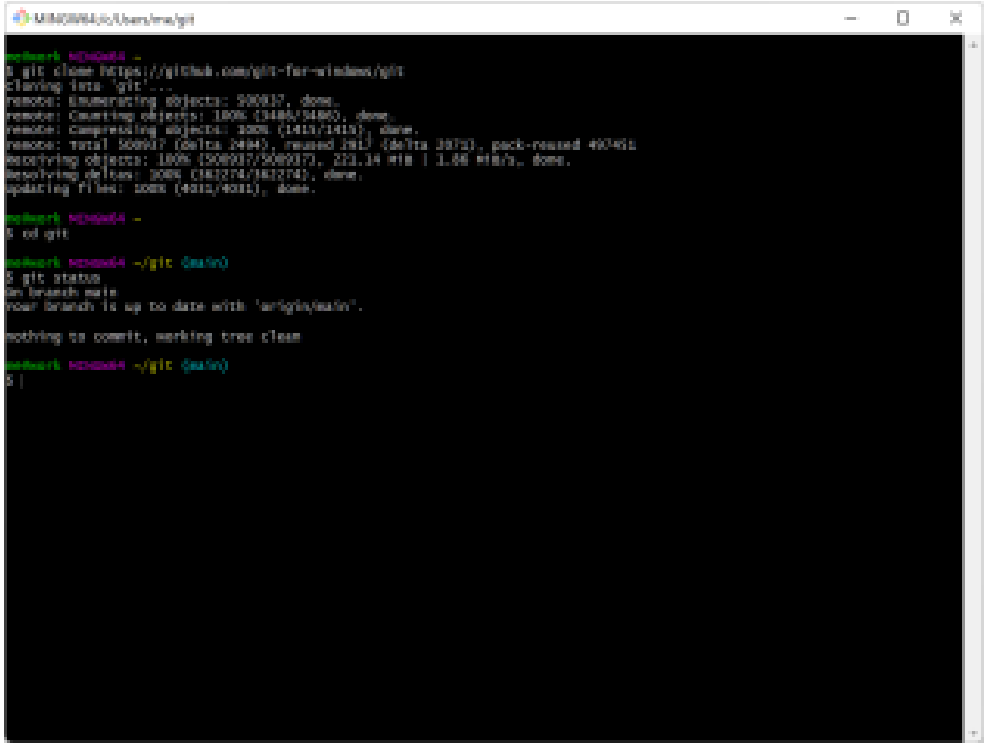
1. Copy and paste the second command



Step 8

1. Open a new tab and test it
2. Type brew update
3. Hit Enter

Installation onWindows:



Git for Windows focuses on offering a lightweight, native set of tools that bring the full feature set of the Git SCM to Windows while providing appropriate user interfaces for experienced Git users and novices alike.
Git BASH
Git for Windows provides a BASH emulation used to run Git from the command line. *NIX users should feel right at home, as the BASH emulation behaves just like the "git" command in LINUX and UNIX environments.

Basics of Git Cont ...

```
Welcome to Git (version 1.9.2-preview20140411)

Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

Bucky@BUCKY-PC ~
$ git config --global user.name "Bucky Roberts"

Bucky@BUCKY-PC ~
$ git config --global user.email "bucky@pornhub.gov"
```

Example1-1

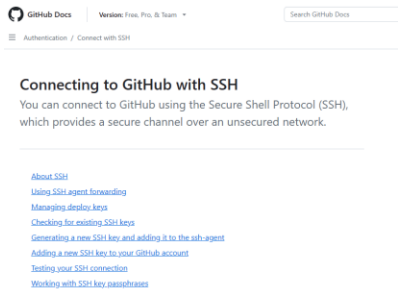
Setup Git local configuration

- Open the command prompt
- Run this command to configure your username and email:

git config --global user.name "Your Username"

git config --global user.email "youremail@email.com"

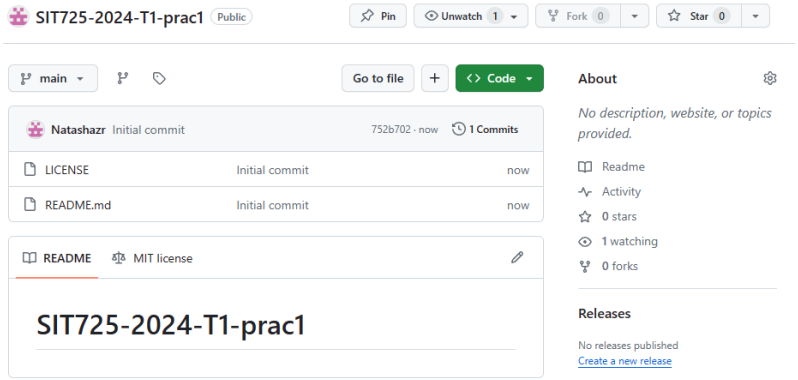
In case you want to generate an SSH Key and add to your git you can do so by following the steps here



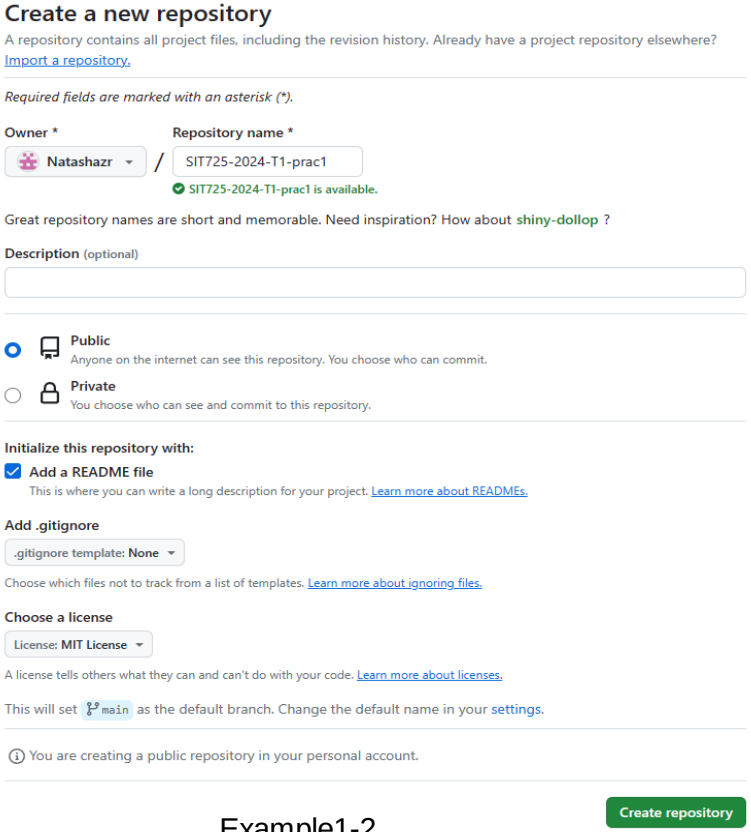
Creating a Git Repo

We need to create a repo online before we start on the system

- Navigate to https://github.com/new and create a repo



Example1-3



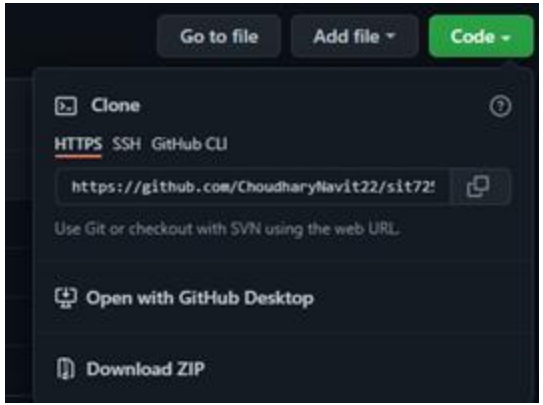
Example1-2

Cloning a Git Repo-MacOS

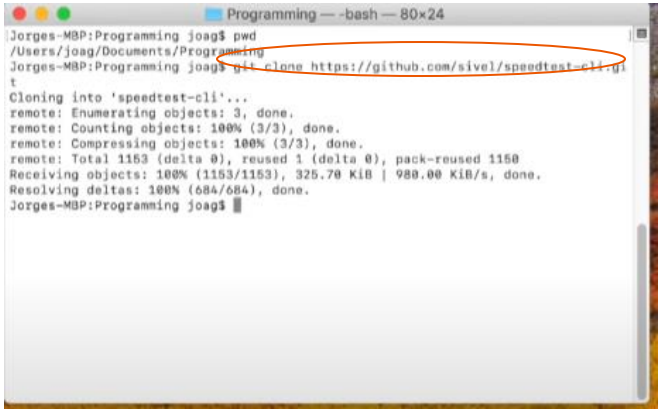
In order to clone a git repo, you simply got to the terminal and use the command

```
$ git clone <link to repo>
```

Link to your repo is present here



Example1-5



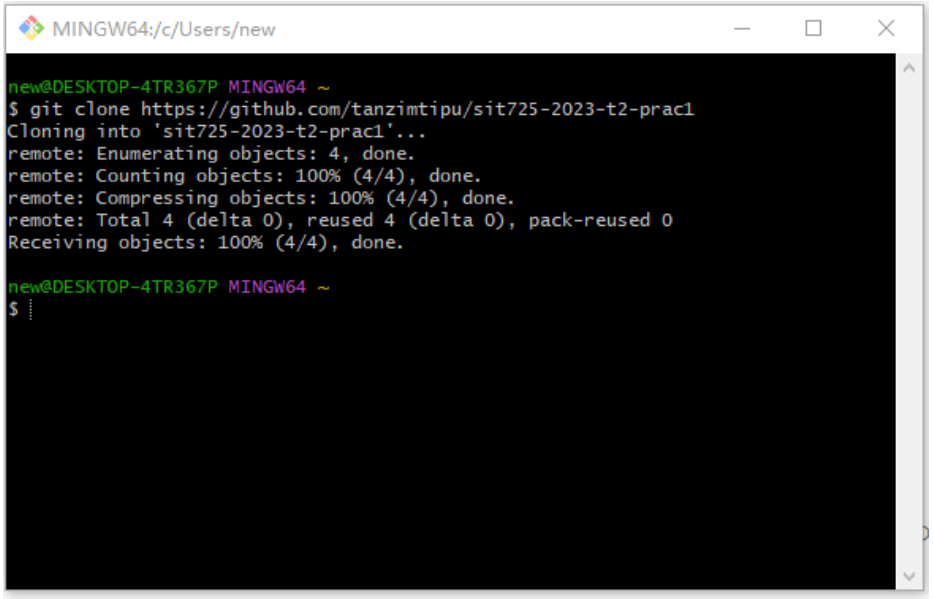
Example1-6

Your password: personal access token

- **In the upper-right corner of any page, click your profile photo, then click Settings.**
- **In the left sidebar, click Developer settings.**
- **In the left sidebar, you see Personal access tokens**

You can also connect using gh:
brew install gh
gh auth login

Cloning a Git Repo-Windows（Git for Windows）

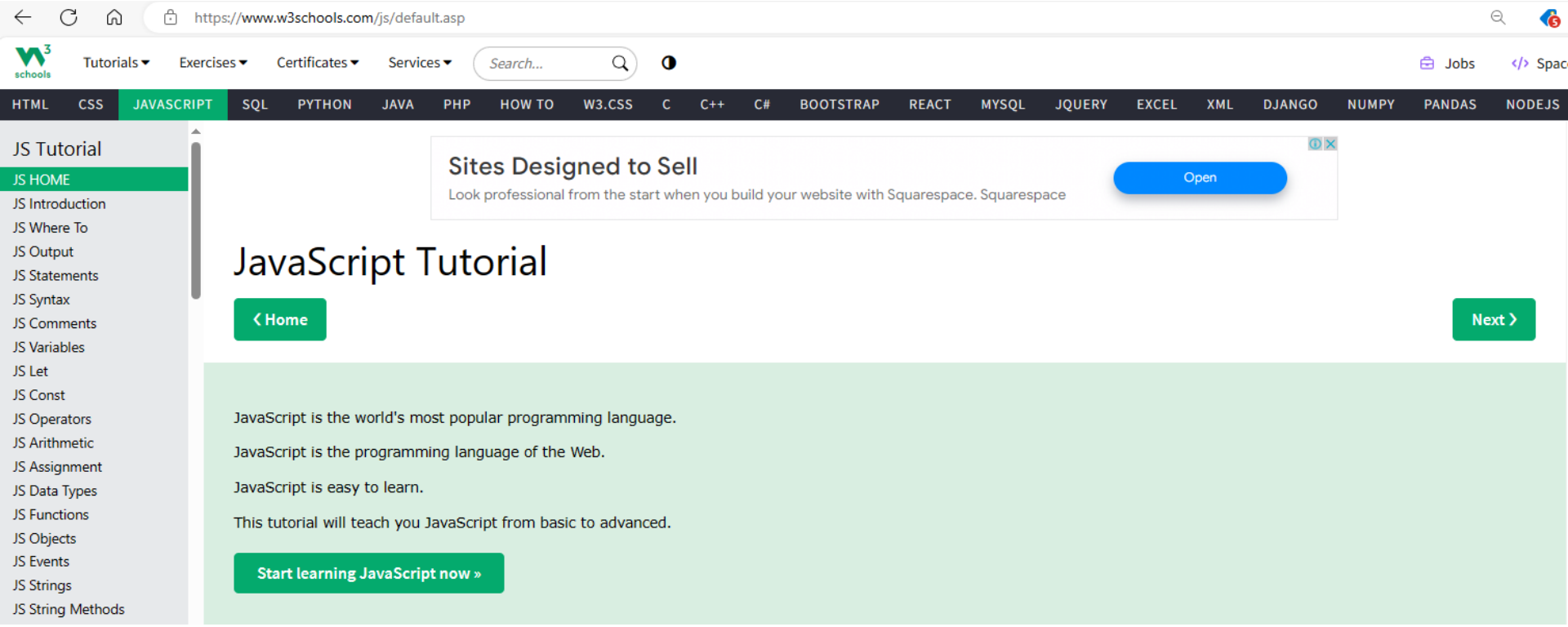


Javascript

JavaScript is a programming language that was first created in 1994 as a way to add functionality and user interaction to a website. JavaScript is the third of the major building blocks of a web page. Without it, we wouldn’t have the dynamic content and usability we expect from modern websites.

JavaScript was created with the intention of being utilised solely on the front end as a mechanism for web developers to add functionality to their pages, and it did so in its early days. The "V8 engine," introduced by Google recently, has improved the speed and usefulness of JS. As a result, interesting new front end JavaScript frameworks were developed and released, as well as Node.js, a mechanism to run JavaScript on a server.

W3schools-Javascript Tutorial

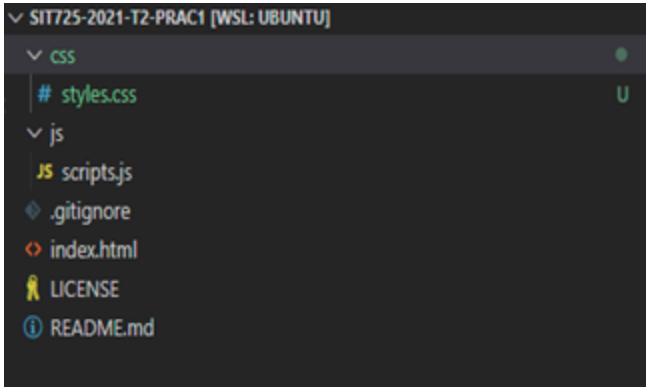


Creating a Basic Javascript application

A simple javascript project has a few files that help us control our javascript application. These files include:

- Index.html: This file holds all the static code that needs to be displayed on the web browser
- styles.css: This file is responsible for styling our web page created by index.html
- scripts.js: This file is responsible to provide the dynamic changes in our static web page being created by the index.html

The folder structure of a simple javascript file looks something like this.

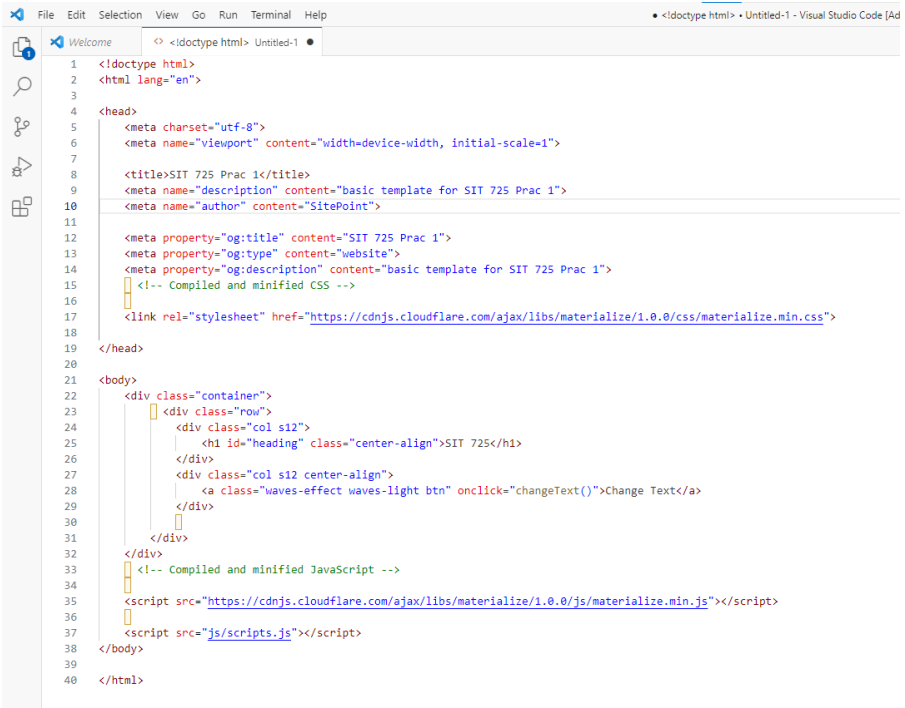


Creating a Basic Javascript application Cont ...

Now let's add some code to our files.
First we start with index.html

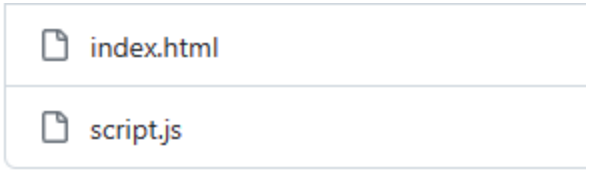
In index.html :

- `<!Doctype html>`
- `<head>,</head>;`
- `<body>, </body>;`
- `</html>`



Creating a Basic Javascript application Cont ...

Then, we create Script.js file
The code can be seen below:




```
function changeText(){
    var textsArray = ["Text 1","Text 2", "Text 3", "Text 4", "Text 5"]
    var number = getRandomNumberBetween(0,textsArray.length - 1)
    console.log("Index: ", number)
    document.getElementById("heading").innerHTML = textsArray[number];
}

function getRandomNumberBetween(min,max){
    return Math.floor(Math.random()*(max-min+1)+min);
}
```

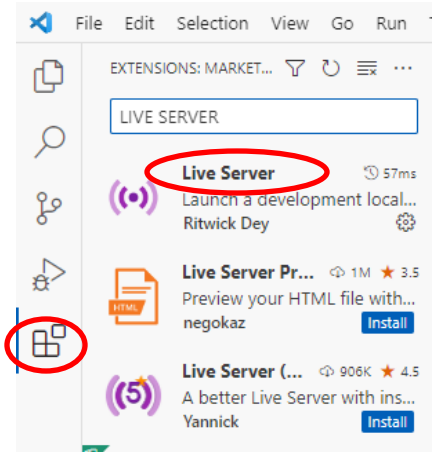
Live Server Extension

Now that we have created our dynamic web page we want to have a look how it looks right, for doing that lets add an extension to our visual studio code that helps us run a live server of our project. You can find the extension here

Once the extension has been installed it will look like something this on your visual studio code .

You can simply press go live and it would open the browser with your current project running. Give it a try!

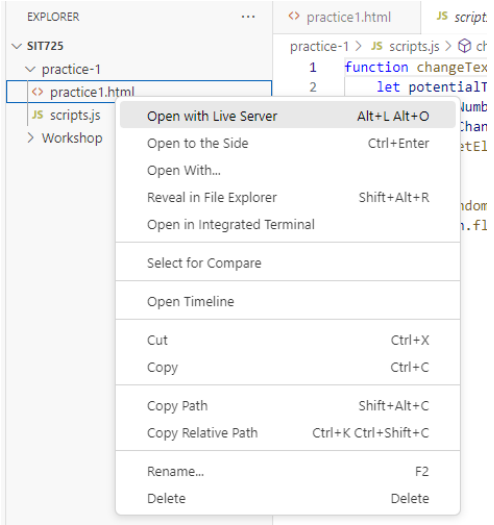
Install extension- Live server in VSC



Live server example

practice1.html

```html
<html>
    <head>
        <title>SIT725</title>
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css">
    </head>
    <body>
        <div class="container">
            <div class="row">
                <div class="col s12">
                    <h1 id="header" class="center-align">SIT725 - Applied Software Engineering</h1>
                </div>
                <div class="col s12 center-align">
                    <a class="waves-effect waves-light btn" onclick="changeText()">Change Text</a>
                </div>
            </div>
        </div>
        <script src="scripts.js"></script>
    </body>
</html>
```

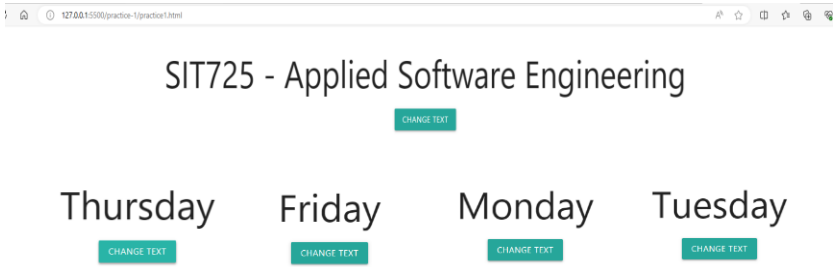


practice1.html | scripts.js practice-1 | scripts.js

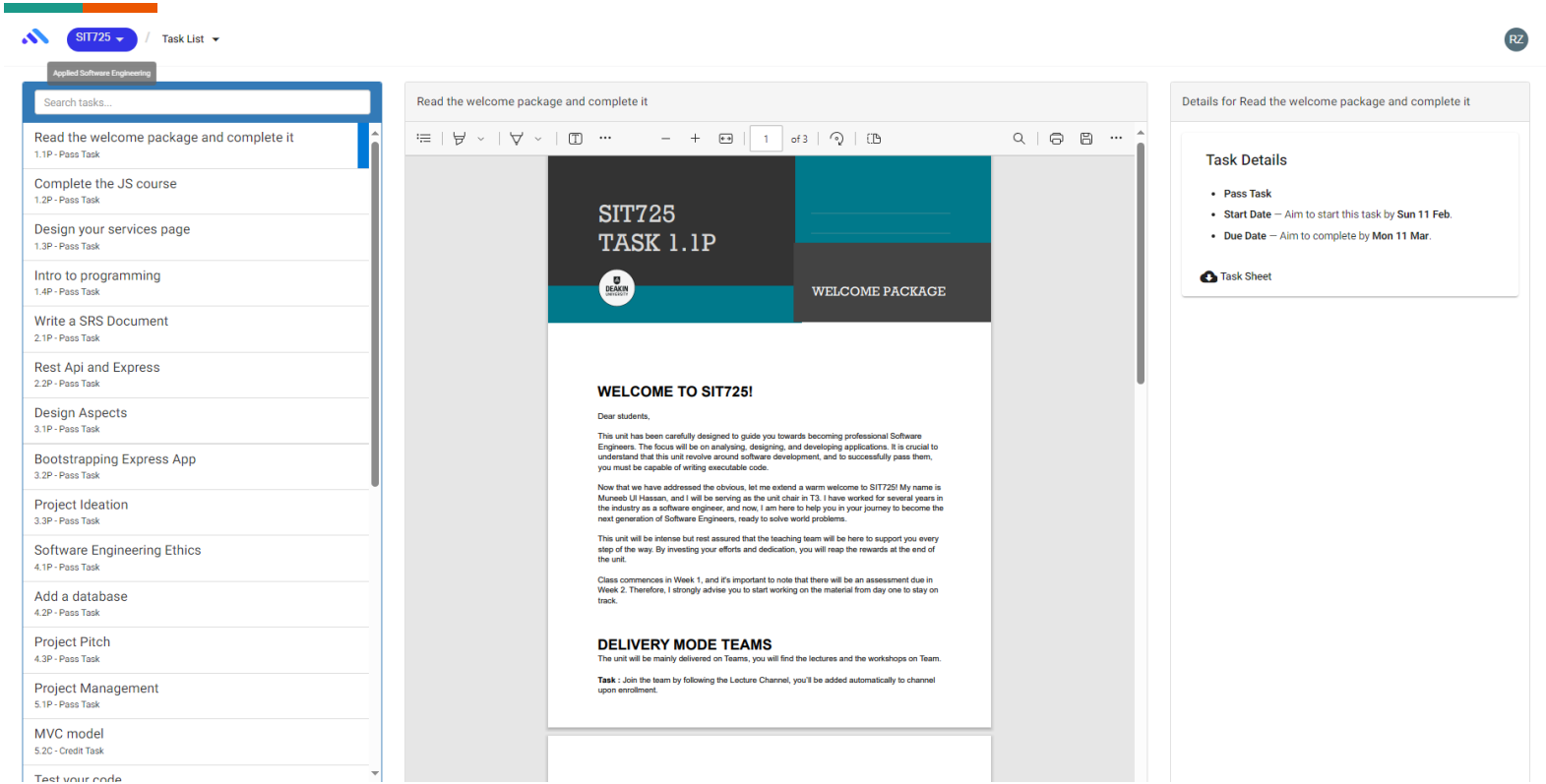
practice-1 > scripts.js > changeText

```js
function changeText() {
    let potentialTexts = ['Monday','Tuesday','Wednesday','Thursday','Friday'];
    let randomNumber = getRandomNumber(0, potentialTexts.length - 1);
    let textToChangeTo = potentialTexts[randomNumber];
    document.getElementById('header').innerHTML = textToChangeTo;
}

function getRandomNumber(min,max){
    return Math.floor(Math.random()*(max-min+1)+min);
}
```



Ontrack Tasks



SIT725 TASK 1.1P

TECH STACK

The technology stack is based on NodeJS, React, HTML and Bootstrap/Materialize. You are expected to have learnt programming prior to joining this unit, if you have never written a line of code before, make sure to start learning now. By Week 3 you should have completed the following, it is a 24hours course that should include most things we are going to be doing this trimester. You need to be proficient from day one. The course is mandatory.

Task1: Complete the following by Week 3 and start from Week 0 if you are not confident in your programming skills, the course is free, log in to LinkedIn learning using your Deakin account. The whole course needs to be completed by the end of week 4, but some tasks will be weekly.

https://www.linkedin.com/learning/collections/6579539778742681602?trk=share_collection_url&accountId=2104084&u=2104084&success=true&authUUID=YInlwdrbTrqN19%2FPh2FCig%3D%3D

Task2:

To successfully submit your practical-related tasks during the trimester, it is essential to create a GitHub account for this unit. Follow the simple instructions provided below to create your account:

1. Visit the GitHub website at www.github.com
2. Click on the "Sign Up" button to begin the account creation process.
3. Enter your preferred username, email address, and a strong password.
4. Select the "Free" plan option,
5. Complete the security verification process, if prompted.
6. Review and accept the terms of service and privacy policy.
7. Click on the "Create account" button to finalize the account creation.

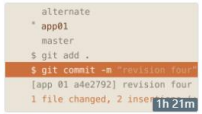
Once you have successfully created your GitHub account, you will be ready to submit your practical-related tasks throughout the trimester.

Full Stack Developer

Created by **Alessio Bonti**

Good complete course

Share Copy



COURSE
Learning Git and GitHub (2015)
By: Ray Villalobos

COURSE
HTML Essential Training (2017)
By: James Williamson

COURSE
CSS Essential Training 1 (2017)
By: Christina Truong

COURSE
Bootstrap 4 Essential Training
By: Ray Villalobos

COURSE
Web Programming Foundations
By: Morten Rand-Hendriksen

COURSE
JavaScript Essential Training (2017)
By: Morten Rand-Hendriksen

COURSE
Node.js Essential Training (2019)
By: Alex Banks

SUBMISSION DETAILS

Once you finished reading this task sheet, leave a comment describing your programming skills, your familiarity with Java script, and your target grade for this unit, then mark this task as "ready to feedback".

SIT725 TASK 1.2P-COMPLETE JS COURSE

Intro

In this unit we will focus mostly on Javascript in the NodeJS flavor to develop our applications. Using online learning resources is very important, also, certificates look really cool on your LinkedIn profile.

Instructions

Please ensure you log in to LinkedIn Learning using your DEAKIN account to avail of the free access to a wide range of courses. It is essential that you complete the course within the allocated time frame. The course duration is approximately 5 hours. I strongly advise against trying to cram the entire course at the last moment, as well as skipping through the content.

Towards the end of the trimester, you will be presented with coding questions that require you to provide on-the-spot answers. Even if you have achieved a High Distinction (HD) in the unit, failing to answer a basic question such as "Can you please explain how the pop function of an array works?" could result in a failing grade. It is crucial to thoroughly understand the concepts and not rely solely on your academic performance.

https://www.linkedin.com/learning/collections/6579539778742681602?u=2104084

Submission details and Delivery

Once completed, download the certificate and upload it on OnTrack

COURSE

JavaScript Essential Training (2017)

By: Morten Rand-Hendriksen

COURSE

Node.js Essential Training (2019)

By: Alex Banks

SIT725 TASK 1.3P-DESIGN YOUR SERVICE

Intro

In this unit you will learn the basics of developing a complex web application. This can become as complex as you want, but ultimately, it will demonstrate your understanding of the technologies being used.

Instructions

As a person in IT, you want to be able to showcase your abilities. Start thinking about "who am I " or "what do I want to be known for" ? And from this, start thinking about what services, ultimately, you want to sell.

If you are going to be a web developer, perhaps you want to build your own portfolio website, if you are a data engineer, you want to show how good you are with data, perhaps store it, analyse it, and show it. If you are in IT security, maybe you want a service that allows people to upload their code to be analysed.

In the coming weeks, you are going to be building your own website, week by weeks during the workshop sessions. Remember, you are to learn the technology, implementing it is up to you.

Submission details

In 200 words or less, describe your grand plan to become the best IT person your field! Don't be shy, use images or sketches, but keep it within one page.

Upload a pdf document to Ontrack

Example

"As a Product manager and Software engineer, I am at selling my expertise as a consultant for prototyping and blockchain. My website will show the projects I have worked on and will also allow users to create an account on my NFT market. Every new user will receive a wallet and will own an automatically uniquely generated NFT art piece."

SIT725 TASK 1.4P-INTRODUCTION TO PROGRAMMING

Intro

Workshops are compulsory, and you must complete them each week. In the Welcome Package (Task 1.1P) you would have provided your Github account. Make sure you have done so, and that the repositories that you make are public from now on. Every Wednesday morning a script is automatically executed and it will download your repository.

It is a machine, so it will not make mistakes, the only reasons it may not be able to download your code is if it unreachable. Also, if you submit after the due date (Sunday) it will be automatically marked as late.

Instructions

In the first workshop you will learn about the following:

- Basic programming
- Git

Submission details and Delivery

- Once you are done, push your code into your repo, giving the repository the following name sit725-2023-t2-prac1 , ultimately this should read as https://github.com/username/sit725-2023-t2-prac1.
- Share your repo link and provide screenshots of the running code as evidence, convert it to .pdf and upload it to the OnTrack.

If you want to be sure, simply clone your repo. If you can clone it, I can clone it 😉

1. You need to follow the instruction of:
   - slides 14,15,16 and 18 to complete Ontrack 1.1p task.

2. In your prac-1, you need to have index.html, Scripts.js, LICENSE, and README.md files. (you can write your own index.html, use the provided Scritp.js code in slide 16 to show the output. The output should be similar to the example on slide18)

3. Test the code, run the page, and do screenshots, convert them in PDF format, and submit evidences into Ontrack system.

Thanks