
ITSE-2192 Fundamental of web Design and Development

Introduction to Git

Lab Zero

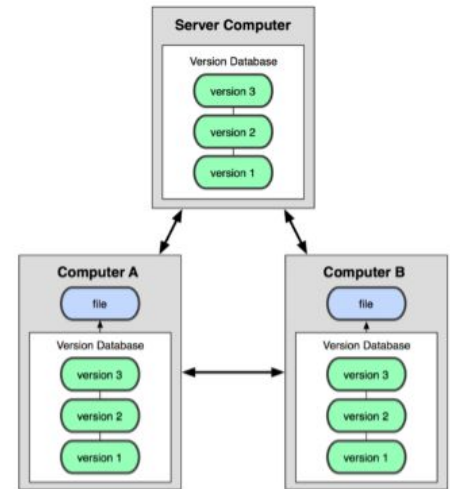
Outline

- Version Control System
- History about git
- How to Install Git
- Working With Git locally
- Working with Github

Version Control System

- Version control is the management of changes to documents, primarily computer programs.
- Also known as revision control or source control
 - Examples: git, mercurial, subversion
- Why version control?
 - Makes working in a team easy!
 - Code without interference
 - Go back to a previous version
 - Integrate code of multiple developer's easily
 - Know who did what, when
- Why Git then ??
 - Distributed
 - Git keeps "snapshots" of the entire state of the project
 - Git generates a unique SHA-1 hash – 40 character string of hex digits, for every commit
 - Refer to commits by this ID rather than a version number

Distributed Model



About Git

- Created by Linus Torvalds, creator of Linux, in 2005
 - Came out of Linux development community
 - Designed to do version control on Linux kernel
- Goals of Git:
 - Speed
 - Support for non-linear development (thousands of parallel branches)
 - Fully distributed – Able to handle large projects efficiently
- (A "git" is a cranky old man. Linus meant himself.)



How to Install Git

Things you'll need:

1. You need Git installed on your system, and you can access it in a UNIX Terminal, either the Terminal on the Mac or Git Bash on Windows.
2. Download Git from the following link:
 - a. Download Link [[Link](#)]
 - b. Installation Process [[Link](#)]

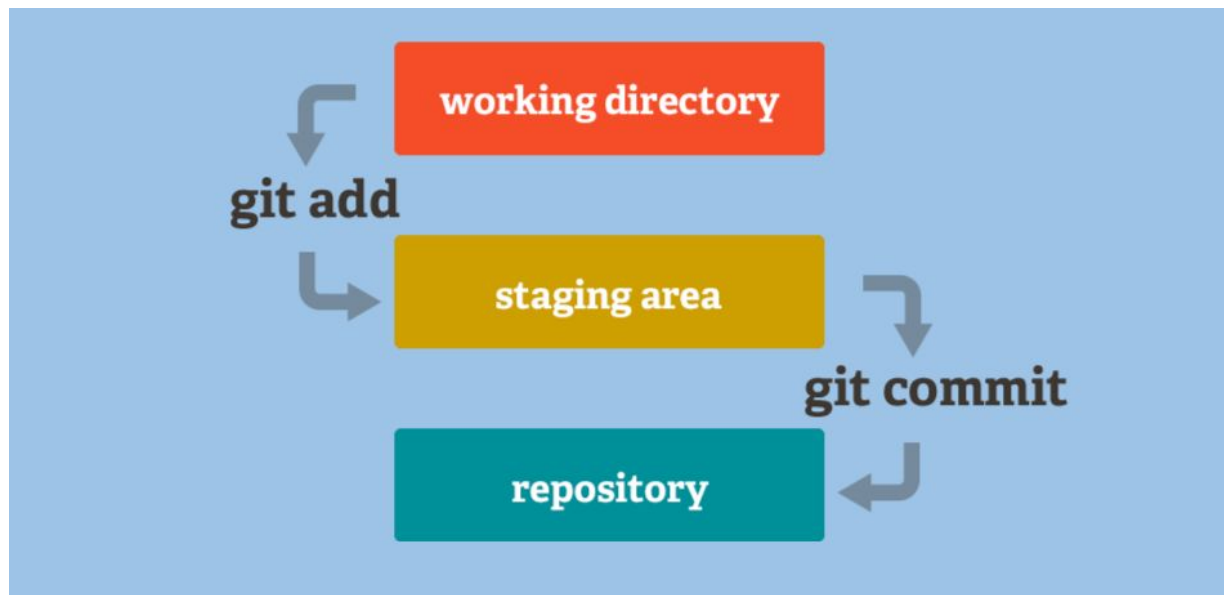
Basics Git Commands

1. Run the below command on the terminal to view about git commands
`git`
2. Run the below command on the terminal to view git version
`git --version`
3. To get help about the git command write the command
`git help init` what do you get??
4. To clear git screen type
`clear` or `ctrl+l`
5. Introduce yourself to git (git config)
`git config --global user.name "your name"`
`git config --global user.email "your mail"`
6. To view the configuration
`git config --list`

Git Model(the work flow)

The Work Flow

- **Modify** files in your working directory
- **Stage files**, adding snapshots of them to your staging area
- **Commit**, which takes the files in the staging area and stores that snapshot permanently to your Git directory



Create and Maintain Local Repository

1. Create a folder named project1 in Other local drive than C: and run the following commands below

Type `pwd` //to view current directory

Type `ls` or `ls -a` (hidden files) //to view content of the directory

Type `cd drive-letter:` //to change drive (d: , e: ,.....)

Type `mkdir project1` //create the directory

Type `cd project1` //change to the directory

Check the Status >>>>Type `git status`what do you get??? Let go ahead.

Create and Maintain Local Repository(Cont..)

1. Create a new repository by running the command
`git init` //This will create a .git directory in your current directory
`git status` what do you get now???????
2. Create a new empty file inside project1 and name it README
`touch file1.txt` //you can create the file manually
`git status` something wrong???
3. Tell git repository track the file to, run the command below to add it.
`git add file1.txt` (or `*.txt` , or `.` to track all files)
`git status`what do you get now???
4. To commit the file to the local repository, issue the commit command
`git commit -m "Your Commit message "`
`git commit` //if you need to write a long message :wq(to save and exit)
`git commit -a -m/-am "Your Message"` //to skip the staging area

View Commit History

1. To see the version history of your repository, run the command below

```
git log
```

```
git log --oneline           //to print on one line
```

```
git log -p -2              // the last two commits
```

```
git log --since=1.weeks    //since last week
```

```
git log --since="2014-05-05" //since specific day
```

```
git log --author="name"    //change made by specific author
```

```
git show 45678789e8789789
```

```
git log --graph --pretty=oneline
```

```
gitk(Windows)/gitx(Mac)
```

Status and Diff

To view status

`git status` or `$ git status -s` (-s shows a short one line version)

- To see what is modified but unstaged:

`git diff`

- To see staged changes:

`git diff --cached`

Undoing Commits and Changes

To return to any point of time

```
git checkout HasID(45678789e8789789 ) or
```

```
git checkout -- filename
```

To unstage a change on a file before you have committed it

```
git reset HEAD filename (unstages the file)
```

To Delete a untracked file

```
git rm <file>
```

To deleted staged file (after you add the file)

```
git rm -f <file >
```

To remove file from staging area (after you add the file)

```
git rm --cached <file >
```

Remote Repository hosted by GitHub

1. create an account on github.com
2. create a repository called fordemo on your account by clicking the “+” sign on your account
3. Now let's push what we have on local to remote repository

```
git remote add origin url
```

```
git push -u origin master
```

4. If there is proxy, you need to set it like the following set

```
git config --global http.proxy 10.90.10.70:8080
```

5. To remove if you mistake account

```
git remote remove origin
```

Remote Repository hosted by GitHub(Cont..)

4. Synchronize the new cloned project and the old

```
git pull
```

5. To pull data without merging

```
git fetch origin
```

6. To pull data with merging

```
git remote -v//to view remote repositories
```

```
git pull "fetch url"
```

7. To Rename the name origin

```
git remote rename origin <anyname>
```

Resource

- Free online book: <http://git-scm.com/book>
- Git tutorial: <http://schacon.github.com/git/gittutorial.html>
- Reference page for Git: <http://gitref.org/index.html>
- Git website: <http://git-scm.com/>
- official Doc : [Doc](#)

ITSE-2192 Fundamental of web Design and Development

Hyper Text Markup Language(HTML)

Lab One

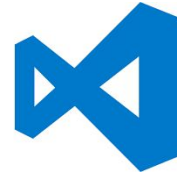
What we will Learn

- After completing this lab :
 - You will be able to work on basic html tags
 - Learn the major difference on HTML5 from earlier versions
 - You will be able to build you personal web app[CV]
 - Hosting in Glthub
- We will go through learning the html by building our personal cv with different pages

Required Software's

- Text Editor [Any one of them]

[VsCode](#) [Recommended for this course]



[Bracket](#)



[Atom](#)



[Sublime](#)



[Check for More](#)

- Browsers
 - **Chrome [Recommended for this course]** , Mozilla , Safari , IE
- Online Version [if you want to work on the cloud]
 - Check this list of editors online : [Link](#)

HTML

- **HTML : Hyper Text Markup Language**

- It is a scripting language but not a programming language
- Current Version : HTML5
- Defines the Structure and Content of a document to be rendered in a browser
- Html Tags
 - `<tagName attributes> Contents </tagName>`
 - `attributes` ⇒ `key = value` pairs , the value can be " "
 - Tag Specific or Global
 - Some tags are **self closing tags** like `
`, `<hr>`

Note : HTML5 do not promote styling using the some tag attributes even though it do not prohibit you from using them , so you are not encouraged to use styling in html tags

HTML Boilerplate

1. Create a Folder Called **MyCv**
2. Open The Folder with VSCode
3. Create a file **index.html** //This file naming is Mandatory for hosting
4. Use The **Emmet** [will use more as we go on the course] >> in VsCode to generate Code
 1. Type **!** And Enter //This will Generate the HTML5 boilerplate
 2. Change the title with your name CV

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Your Name Cv</title>
</head>
<body>
</body>
</html>
```

Working with Text Elements and Line

- Heads
 - `<h#> # 1 – 6` [Block element with closing tag]
 - This element create a heading text with large font
 - **Attributes** : align(obsolete)
- Paragraphs
 - `<p>` [Block element with closing tag]
 - This element Creates a paragraph
 - **Attributes** : align(obsolete)
- Horizontal line
 - `<hr>` [Block element and self closing tag]
 - **Attributes** : size , color, width, align

Working with Text Elements and Line(ctnd..)

- Inside the body tag add information Regarding yourself
 - Name : h1
 - Occupation , Place of Work : h3
 - Bio : p
 - Horizontal line for beauty and separation : hr

```
<h1>Your Name</h1>
<h3>Student , Addis Ababa institue of Technology</h3>
<p>Hi there , </p>
<p>I am Your Name , i am expert <strong>Java</strong> Developer
who has been working on Java since <em>5 </em></em5>years. I am
passionate about working in diffrent types of porgaming language
</p>
<hr>
```

Working with List

- Unordered List
 - ` ` [Block element with closing tag]
 - This element create a list with different kinds of bullets
 - **Attributes** : `type(circle, disc, square, triangle)`
- Ordered List
 - ` ` [Block element with closing tag]
 - This element create a list with different kinds of bullets
 - **Attributes** : `type(a,A, i, I , 1(default)) , start, reversed`
- We can create a complicated list with nesting

Working with List(ctnd..)

- Below the hr tag add the following list
- Hobbies and Interest : h3
 - List of Hobbies and Interest: ul [Feel free to change with your own]

```
<h3>Hobbies and Interest</h3>
<ul>
  <li>Sport : I like to watch and play
    <ul>
      <li>Football</li>
      <li>BasketBall</li>
      <li>VollyBall</li>
    </ul>
  </li>
  <li>Reading Books
    <ul>
      <li>Educational Mostly Programing Books </li>
      <li>Fictions specialy of literatuer
        <li>Books that impress me the most :
          <ul>
            <li>One Love by Mr. Hadis Almayhu</li>
            <li>Black Magic by Mr. Samuale A.</li>
          </ul>
        </li>
      </li>
    </ul>
  </li>
</ul>
<hr>
```


Working with Images

- Image
 - `<` [Block element with self closing tag]
 - This element allow as to display image
 - **Attributes** : src(mandatory) , hight, width, alt, more
- HTML5 Options
 - `<figure> <figcaption> </figcaption></figure>`
 - **Attributes** : Only Global attributes

Working with Images (ctnd..)

- Below the body tag/ above the heading information add your picture
 - Create folder asses > images
 - add a picture : ``
 - add additional info incase browser did not display it : `alt = "Your Name Profile image "`
 - **url** ☐ **local** or **http://** ?? Relative or absolute path

```

```

Working with Tables

- Tables
 - `<table> <tr> <th> <td>`[Block element with closing tag]
 - This element create a table
 - **Attributes** : cellpadding, cellspacing , border , bgcolor, align
- HTML5
 - `<thead> <tbody> <tfooter> <caption>, <col>, <colgroup>`
 - To show the structural components of a table

Working with Tables (ctnd..)

- Below Hobbies and interest lets add skill with the help of tables
 - Add horizontal line ; hr
 - Add the text **skills** : h3
 - Add your table with one row and two/three skill sets of your own
 - Keyboard shortcut for **emoji** : Window : Win + ./; , Mac : Command + Control + Space

```
<hr>
<h3>Top Skills</h3>
<table cellpadding="10">
  <tr>
    <td>Java for Web and Android </td>
    <td>★★★★★</td>
  </tr>
  <tr>
    <td>HTML, CSS and Javascript</td>
    <td>★★★★★</td>
  </tr>
</table>
```

Working with Tables (ctnd..)

- Update the Header part with table
- Add table with two cell [td]
- Place the **picture in one cell** and **the other content in other cell**

```
<table cellpadding="10">
  <tr>
    <td>
      
    </td>
    <td>
      <h1>Your Name/h1>
      <h3>Student , Addis Ababa institute of Technology</h3>
      <p>Hi there , </p>
      <p>I am Your Name , i am expert <strong>Java</strong> Developer who has been
      working on Java since <em>5 </em></em5>years. I am passionate about working in
      different types of programming language </p>
    </td>
  </tr>
</table>
```

Working with Tables (ctnd..)

- Add a new table at the top of skill [for you to work with]
- **Note :** Add this element above **Top Skills**
 - Add hr
 - Add h3 : Work Experience
 - Add a row with two cells with the header of “Date” and “Job Title” : th
 - Add a row with two jobs you have worked on [just assumption]
 - Apply the HTML5 structural components : thead , tbody

Working with Hyperlink

- Hyperlink
 - `<a>` [inline element with closing tag]
 - This element create a hyperlink between Documents
 - **Attributes** : href(mandatory), target, [download](#), [More](#)
- Other Document relationship tags
 - `<base>`, `<link>`, `<script>`

Working with Hyperlink(ctnd..)

- Above the Head lets create navigation
 - Create Horizontal line below and above the header info: hr
 - Create an empty **contact.html** file on the same level as index
 - Create two links inside the horizontal line
 - Copy the code from index then remove everything in the body except the header part

```
<hr>  
<a href="index.html">Home</a>  
<a href="contact.html">Contact</a>  
<hr>
```


Working with Hyperlink(ctnd..)

- Creating Bottom Navigation (for you to do)
 - Create a bottom navigation with hr surrounded for nice separation on the bottom : a
 - The links should go to your social media : Facebook , tweeter, YouTube , ...

Working with Form

- User Input forms
 - `<form>` `<input>` [inline element with self closing tag]
 - This element create a form with different fields
 - Types : text , password , ..
 - **Attributes**[Form]: , method, action , [more](#)
 - **Attributes**[input]: disable, name , value
- HTML5
 - New types : email , date , time , tel, number, url
 - **Attributes**[Input] : required ,autocomplete, , pattern, [more](#)

Working with Form(ctnd..)

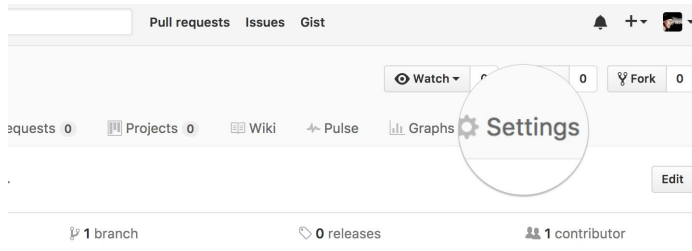
Inside the contact.html add information for receiving inputs from user

- Add Text “ Contact Me With the following Information” : H3
 - Add the form ⇒ action: mailto:" your mail", method: post , enctype: utf-8
 - Add the form with **input** : text , email , submit , with **label**
 - add **text area** : <textarea>
 - add validation **required**

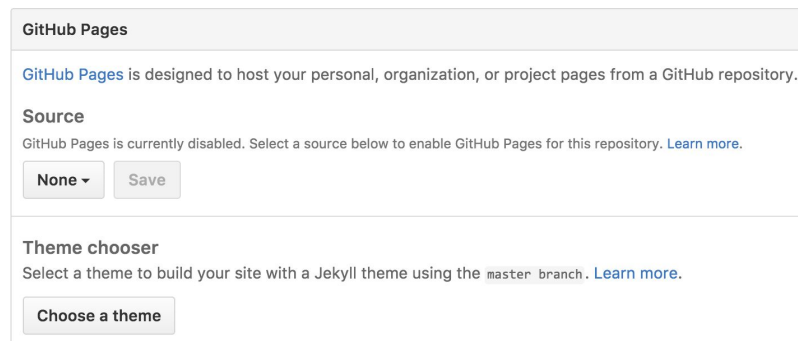
```
<form action="mailto:yourmail" method="POST" enctype="utf-8">
<label for="name">First Name: </label>
  <input type="text" name="fName" id="fName" required><br>
  <label for="name">Last Name: </label>
  <input type="text" name="lName" id="fName" required><br>
<label for="email">Email: </label>
<input type="email" name="email" id="email" required><br>
Leave Your Comment here ..<br>
<textarea name="comment" id="comment" cols="30" rows="10"></textarea>
<br>
<input type="submit" value="Submit">
</form>
```

Hosting in Github

1. Make sure you initialize , and perform a commit operation locally
2. Perform a git push to a repo called **"MyCv"** or any name you like
3. Go Head to the repo setting tab



4. Scroll down to the GitHub Pages section. Press source and choose the master branch.



Miscellaneous Changes

- To make the website more interesting :
 - Adding | between the Navigation Items for good view
 - Adding a rounded image [[Online Link](#)] and updating the the image
 - Adding one **br** in each form element for adding more space between the form elements
 - Adding space between label and the element using ** ** element
 - Adding footer that shows the developer , below the last table
 - To Generate Copy Symbol : **©**
 - For More : [Link](#)

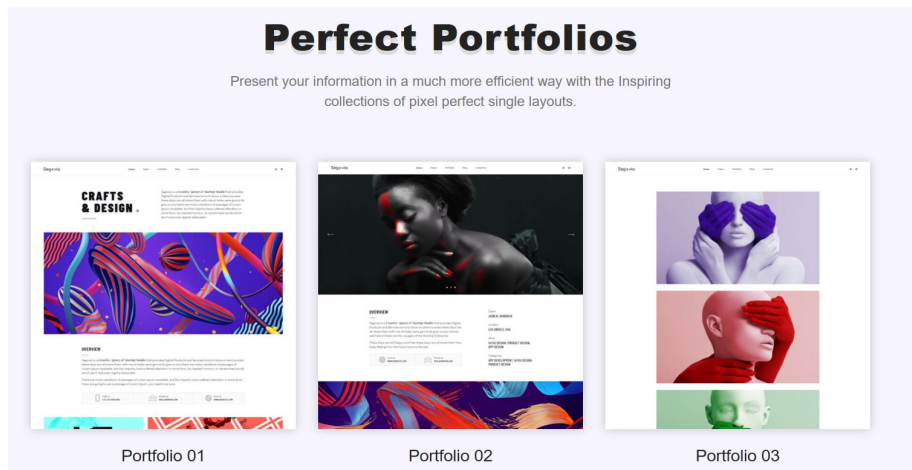
```
<hr>
```

```
<p align="center">Developed By Your Name. &copy;2020</p>
```

```
<hr>
```

Exercise (10pt)

1. Add icon on the head part using **link** tag inside the **head tag** : [Link](#)
2. Update the contact form to be displayed inside a table where each form take a new row [and **Label : Form** Element with two cells in a row]
3. Add more [html5 input elements](#) to the contact page
4. Use the [pattern attribute to validate](#) first and last name name , password
5. Add Gallery page [Update the navigation on the top to have **Gallery**]
 - a. Three images in a row
 - b. Min of 3 rows [You can use category to show what the images are about at the top of each row]
 - c. use hr, figure with caption , and table [with table attributes to make it good looking]



6. **Bonus++** : [Embed Map](#) on the About page

Resource for Further Working

1. [MDN Downloadable Cheat Sheet](#)
2. [HTML5](#)
3. [Client Side Form validation](#)

ITSE-2192 Fundamental of web Design and Development

Cascading Style Sheet(CSS)

Lab Two

ITSE-2192 Fundamental of web Design and Development

Bootstrap 3 Basics

— Lab Three —

ITSE-2192 Fundamental of web Design and Development Javascript Basics

Lab Four

ITSE-2192 Fundamental of web Design and Development **Advanced Javascript**

Lab Five

ITSE-2192 Fundamental of web Design and Development

Node Js Basics

Lab Six

ITSE-2192 Fundamental of web Design and Development Node Js in Depth

Lab Seven
