Write your name below and indicate your role,

Project Manager (PM), Recorder (R)

Name ____ Role _________

Name ____ Role _________

GitHub Workshop Part 1

Your Tasks (Mark these off as you go)

- □ Define key vocabulary
- □ Create a GitHub account
- □ Have Ms. Pluska check off Define key vocabulary and Create a GitHub account before you continue
- □ Install Git
- □ Set the Git commit username and email
- □ Have Ms. Pluska check off Install Git and Set the Git commit username and email before you continue
- □ Fork a repository
- □ Clone a repository
- □ Commit to a repository
- □ Receive credit for the group portion of this lab

□ Define key vocabulary

- Open up your browser and search for definitions to the following terms as they relate to “Git”. For example you could type “Git vocabulary”.
- Work with your partner to write definitions for the following terms.

Git	
GitHub	
repository	
local	
remote	
commit	
push	

clone	

fork	

pull	

origin/master	

□ Create a GitHub account

Navigate to the following address,

https://education.github.com/students

Locate the “Get Benefits for Students” button and click on it.

Locate the “Create an account” link and click on it.

Create your account when prompted

□ Have Ms. Pluska check off Define key vocabulary and Create a GitHub account before you continue

STOP

Before you continue have Ms. Pluska check off Define key vocabulary and Create a GitHub account

Do not continue until you have Ms. Pluska’s (or her designated TA’s) signature ____________

□ Install Git

Locate the instructions that apply to your OS to install Git. At this point, do not worry about what the code means – you will learn more about this as the year progresses. **Note**: If you are using a school computer, you will be prompted for a password. The default password for the school computers is “password”.

Linux

Open the terminal by typing “ctrl-alt-t”.
In the terminal window type the following commands. After each commend click enter.

```
sudo apt-get update
sudo apt-get install git
```

Windows

Navigate to https://gitforwindows.org and download and install git

Mac

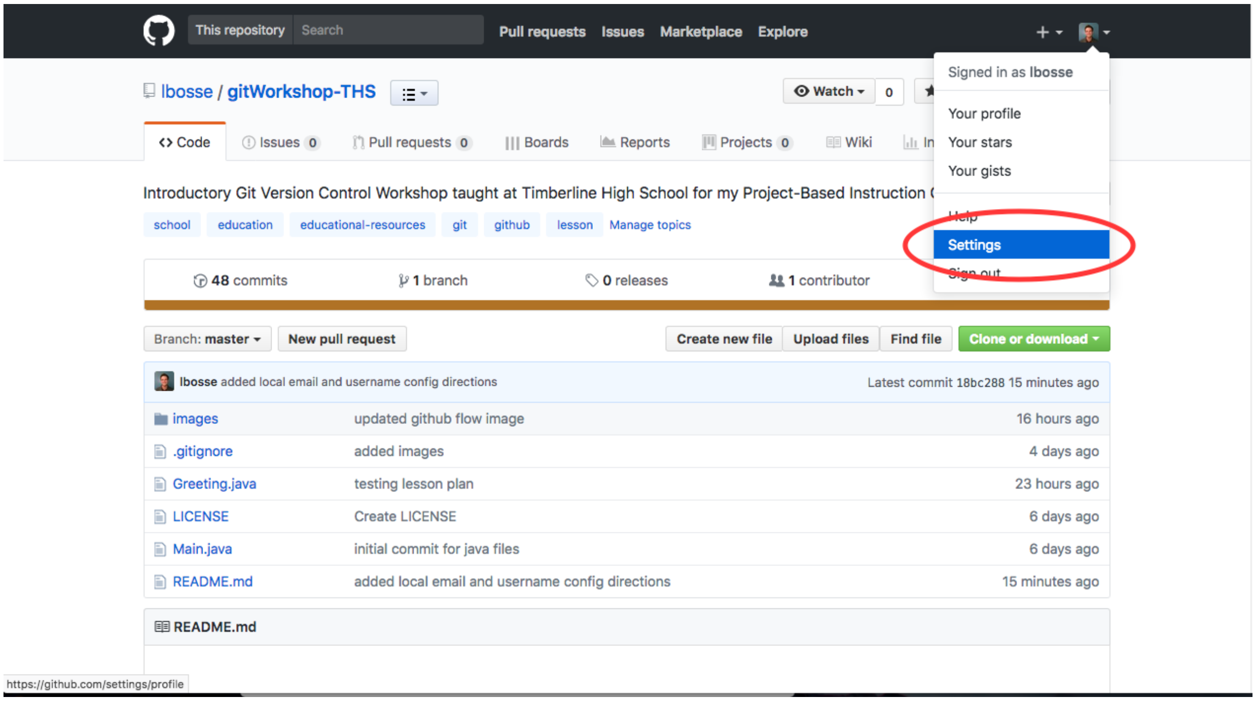
Open a Terminal on your Mac. Now, type the following command into your Terminal.

```
git --version
```

If you don’t have git installed already, it will prompt you to install it.

□ Set the Git commit username and email

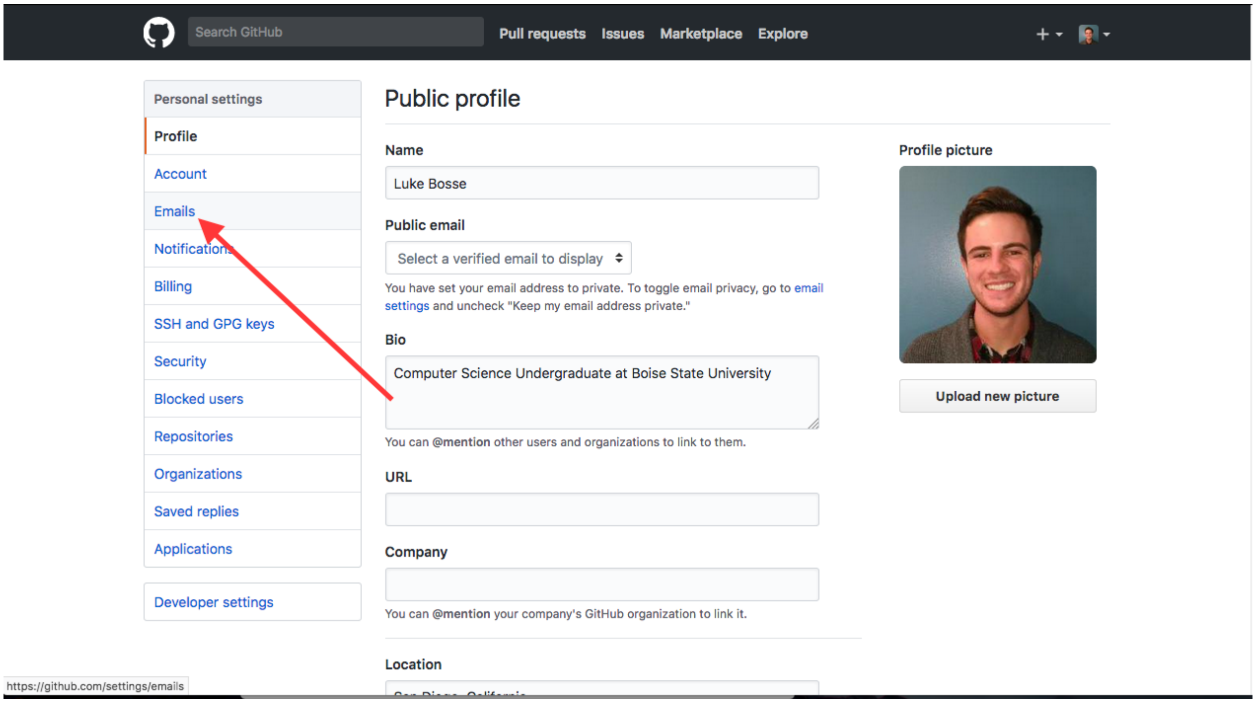
Anyone who has access to your repository can see your commits and the email address associated with them. Since we're using the free account right now, we need to use the no-reply email address that GitHub gives you as your commit email in order to stay safe online. To find it, go to the Email page under your account's Settings page and look at the "Primary Email" section below. Below are some pictures to help you find it:



Introductory Git Version Control Workshop taught at Timberline High School for my Project-Based Instruction (

lbosse added local email and username config directions — Latest commit 18bc288 15 minutes ago

images	updated github flow image	16 hours ago
.gitignore	added images	4 days ago
Greeting.java	testing lesson plan	23 hours ago
LICENSE	Create LICENSE	6 days ago
Main.java	initial commit for java files	6 days ago
README.md	added local email and username config directions	15 minutes ago



Public profile

Name: Luke Bosse

Public email: Select a verified email to display

You have set your email address to private. To toggle email privacy, go to email settings and uncheck "Keep my email address private."

Bio: Computer Science Undergraduate at Boise State University

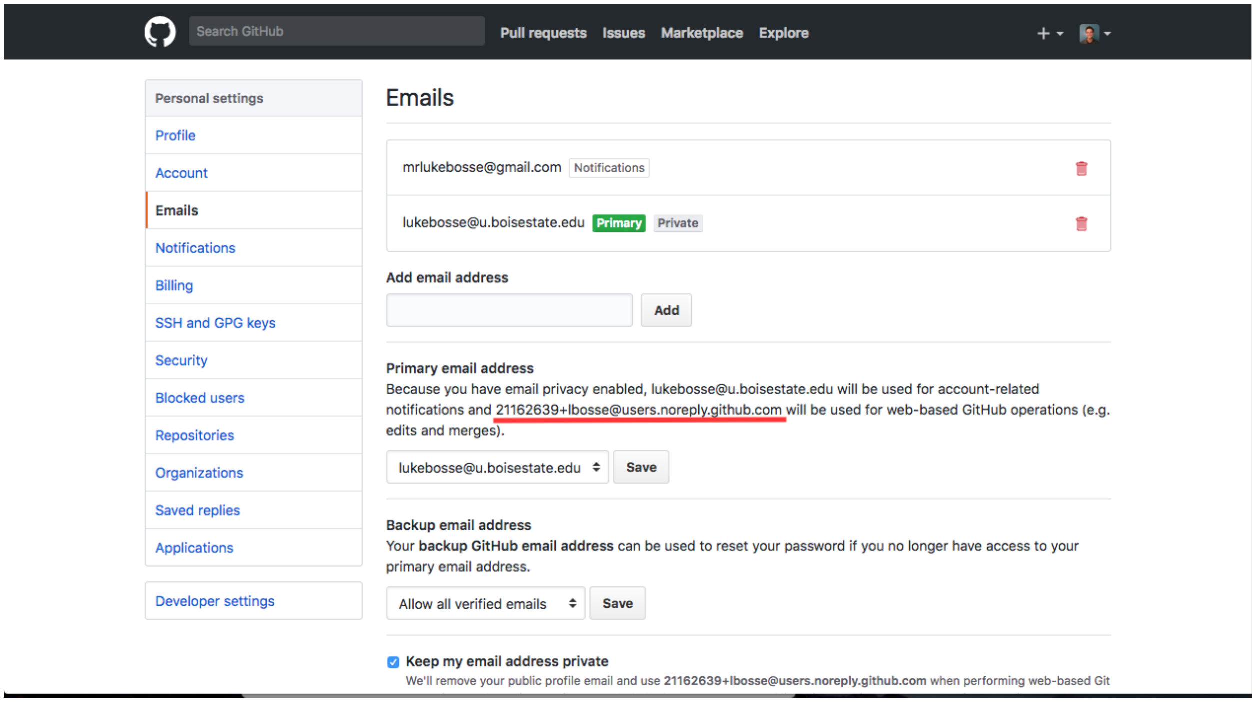
You can @mention other users and organizations to link to them.

URL:

Company:

You can @mention your company's GitHub organization to link it.

Location: San Diego, California



Emails

mrlukebosse@gmail.com Notifications

lukebosse@u.boisestate.edu Primary Private

Add email address

Primary email address

Because you have email privacy enabled, lukebosse@u.boisestate.edu will be used for account-related notifications and 21162639+lbosse@users.noreply.github.com will be used for web-based GitHub operations (e.g. edits and merges).

lukebosse@u.boisestate.edu

Backup email address

Your **backup GitHub email address** can be used to reset your password if you no longer have access to your primary email address.

Allow all verified emails

☑ **Keep my email address private**

We'll remove your public profile email and use **21162639+lbosse@users.noreply.github.com** when performing web-based Git

- Once you locate the no-reply email, highlight it and copy it to your clip board (ctrl-c).
- Now open your command line and type the following. Replace “<your-name-here”>” with your user name. Replace “<your-email-here>” with the email you copied to your clip board. To paste in the command line type ctrl-shift-v. NOTE: Do not include quotes. Do not include < or >.

```
git config --global user.name “<your-name-here>”
git config --global user.email “<your-email-here>”
```

```
git config --list
```

□ Have Ms. Pluska check off Install Git and Set the Git commit username and email before you continue

Before you continue have Ms. Pluska check off Install Git and Set the Git commit username and email

Do not continue until you have Ms. Pluska’s (or her designated TA’s) signature ___________

□ Fork a repository

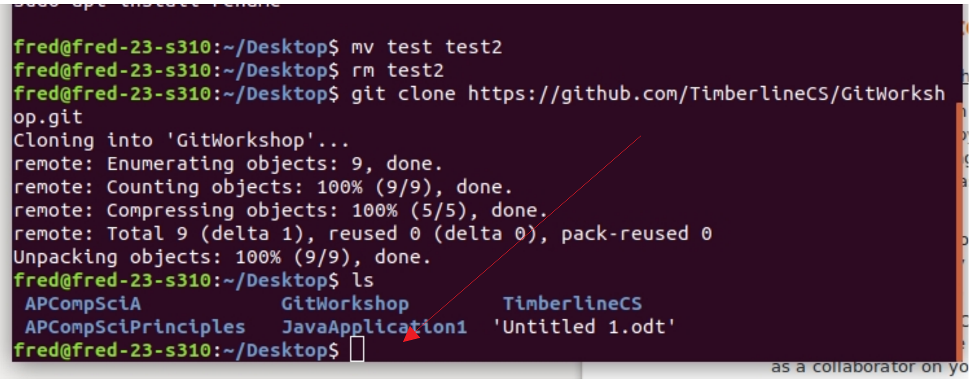
- Navigate to https://github.com/TimberlineCS/GitWorkshop.git
- Click the "Fork" button in the upper right hand corner of the screen. This will create your own personal copy of the repository in your account, where you can modify it freely without changing the code in my repository. This is often used when using someone else's code as a starting point for your own project or for proposing changes to their project.
- You should now see your own copy of this repository on your Github profile page! Click on the repository name to go to it's page.
- Click on the "Settings" tab right underneath the "Fork" button to go to the settings page. Then click the "Collaborators" button on the menu on the left side of the settings page. Use the search box to search for my username (**hpluska**) to add me as a collaborator on your repository. This will allow me to see your code and assess whether or not I did a good job teaching this lesson :)
- Click on the "Code" tab on the right side of the screen to go back to your files. You'll see a lot of buttons and options on this page! We'll get to those shortly. For now, click on the **"Clone or Download"** button and copy the link it gives you. You can also download it as a .zip file, but we're going to do things the cool way ;)

□ Clone a repository

Now that you've got the ".git" link to your repository, open up your terminal (ctrl-alt-t) and navigate to your desktop. Once you're there, type the following command in the terminal to clone the repository for this lab,

```
git clone https://github.com/TimberlineCS/GitWorkshop.git
```

Now list the contents of your desktop (ls). If all went well you will see the GitWorkshop folder.



```
sudo apt install rename

fred@fred-23-s310:~/Desktop$ mv test test2
fred@fred-23-s310:~/Desktop$ rm test2
fred@fred-23-s310:~/Desktop$ git clone https://github.com/TimberlineCS/GitWorksh
op.git
Cloning into 'GitWorkshop'...
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 9 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (9/9), done.
fred@fred-23-s310:~/Desktop$ ls
 APCompSciA            GitWorkshop       TimberlineCS
 APCompSciPrinciples   JavaApplication1  'Untitled 1.odt'
fred@fred-23-s310:~/Desktop$ 
```

□ Commit to a repository

- Navigate into the GitWorkshop folder you just cloned and locate the README.md file. Open this file in gedit,

```
gedit README.md
```

- Answer the prompts then save this file and close it.

Now, time for the good stuff! When we want to save, or "push" our code to our remote repository (the repository on Github you justed forked), we first need to record our changes by "commiting" the changes we made to our code. This whole process takes four quick steps:

- Run the command git status in your GitWorkshop folder. This will show you both changes to "tracked" (files already being watched by git) and "untracked" (new) files. You should see the following message from this command:

```
On branch master
Your branch is up to date with 'origin/master'.


Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)


        modified:   README.md


no changes added to commit (use "git add" and/or "git commit -a")
```

- Git is telling us that changes to README.md are not staged for commit. "Staging a file for commit" means that we tell Git we want to record changes to that file. Add the changes you want to record with your commit by running the following command,

```
git add README.md
```

NOTE: You can add changed files individually by specifing the file name like we did above. If you have lots of changed files the command, “git add .” will stage all the changes.

- Now, make your commit and record your changes by running the the following command where <message> is a message describing the changes. These messages should be short (not much more than a sentence long) and describe what was accomplished. An example of a good commit message would be: "Completed README prompts”

```
git commit -m <message>
```

- Now, you're finally ready to push your code to your Github repository! Run the following command. When prompted type your GitHub username and password.

```
git push origin master
```

- Run the following command.

```
git status
```

If you see the following message you have successfully updated your first GitHub repository! Congratulations!

```
On branch master
Your branch is up to date with 'origin/master'.

Nothing to commit, working tree clean
```

□ Receive Credit for the group portion of this lab

Switch roles, and repeat this workshop with your partner.

STOP

Before you submit your lab have Ms. Pluska check Commit to a repository.

Do not continue until you have Ms. Pluska’s (or her designated TA’s) signature ____________