Sri Lanka Institute of Information Technology

IT Project

Semester 2 – Year 2 – 2017

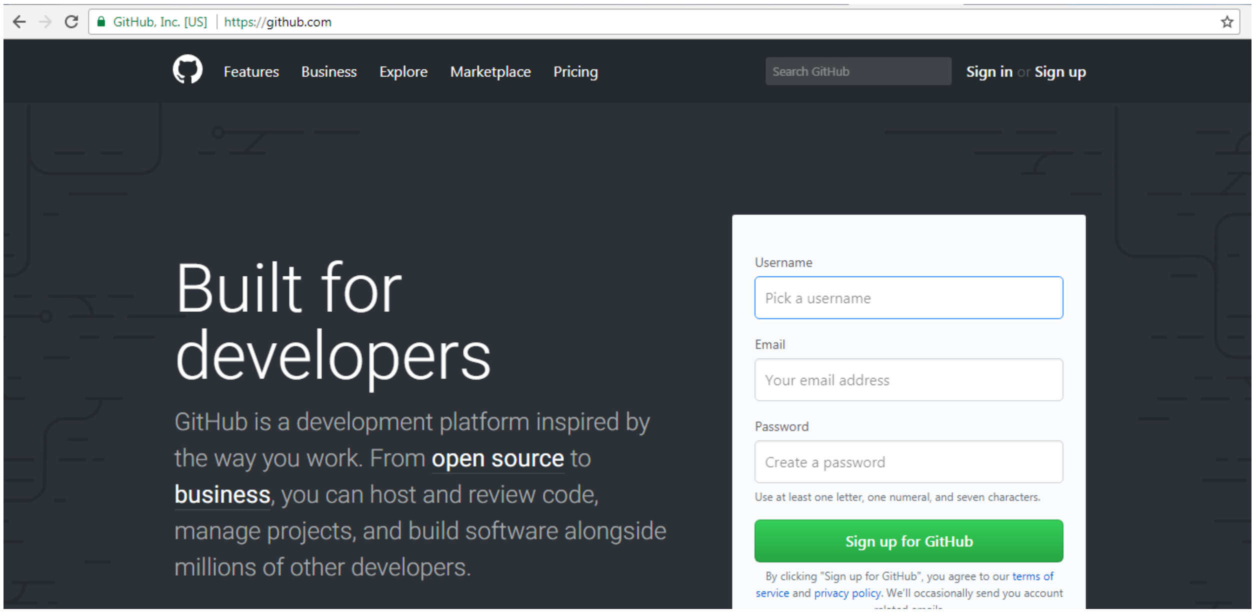
Practical 2

Introduction to Git/GitHub – Version Control System

- Git is a version control system (VCS) for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source code management in software development.
- GitHub is a web-based Git or version control repository and Internet hosting service. It is mostly used for code. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features.

01. Sign up for GitHub using the following URL.

URL : https://github.com/



Built for developers

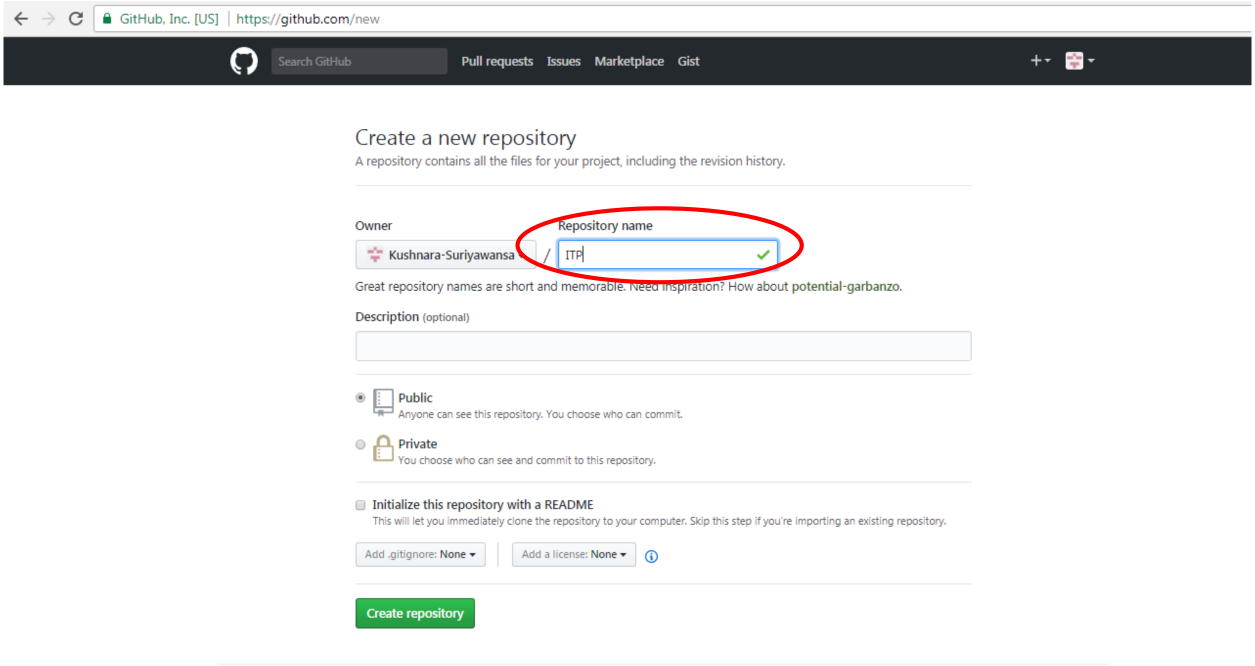
GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside millions of other developers.

Use at least one letter, one numeral, and seven characters.

By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account

Note: You have to remember the username, email address and password used here.

02. Start a project and create a repository. Provide a repository name in the given textbox.



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: Kushnara-Suriyawansa / Repository name: ITP

Great repository names are short and memorable. Need inspiration? How about potential-garbanzo.

Description (optional)

● Public
Anyone can see this repository. You choose who can commit.

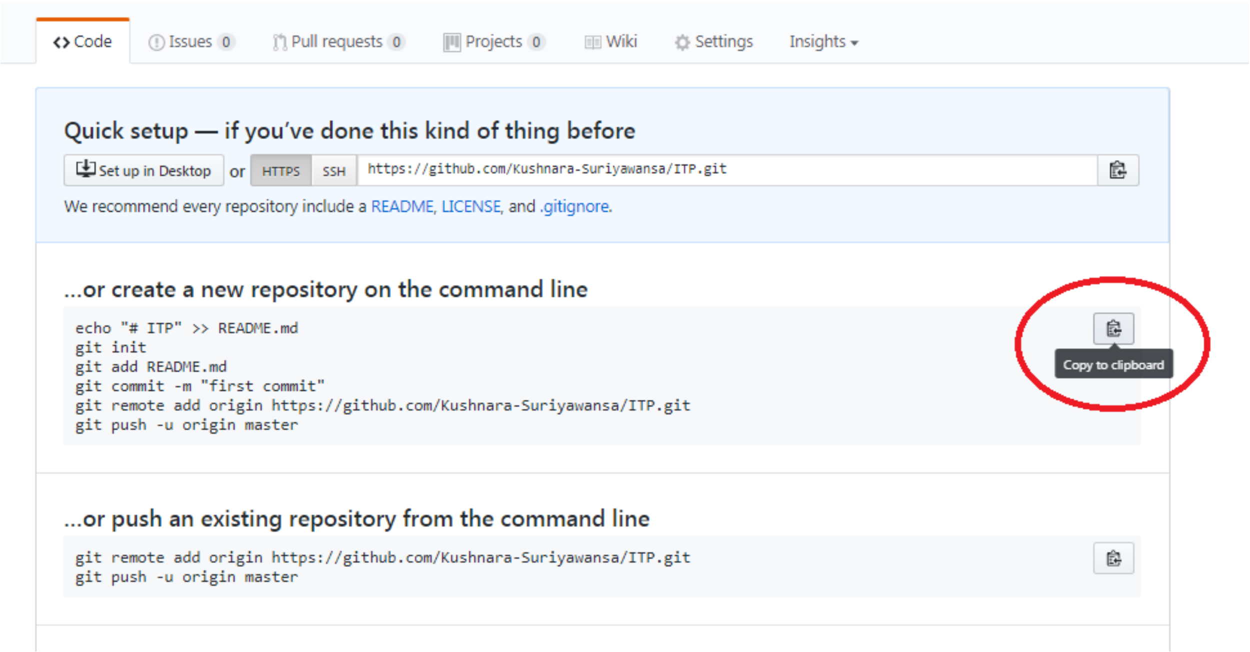
○ Private
You choose who can see and commit to this repository.

□ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None

03. Create a new repository on the command line.

- *Copy the commands using the highlighted button*



Quick setup — if you've done this kind of thing before

https://github.com/Kushnara-Suriyawansa/ITP.git

We recommend every repository include a README, LICENSE, and .gitignore.

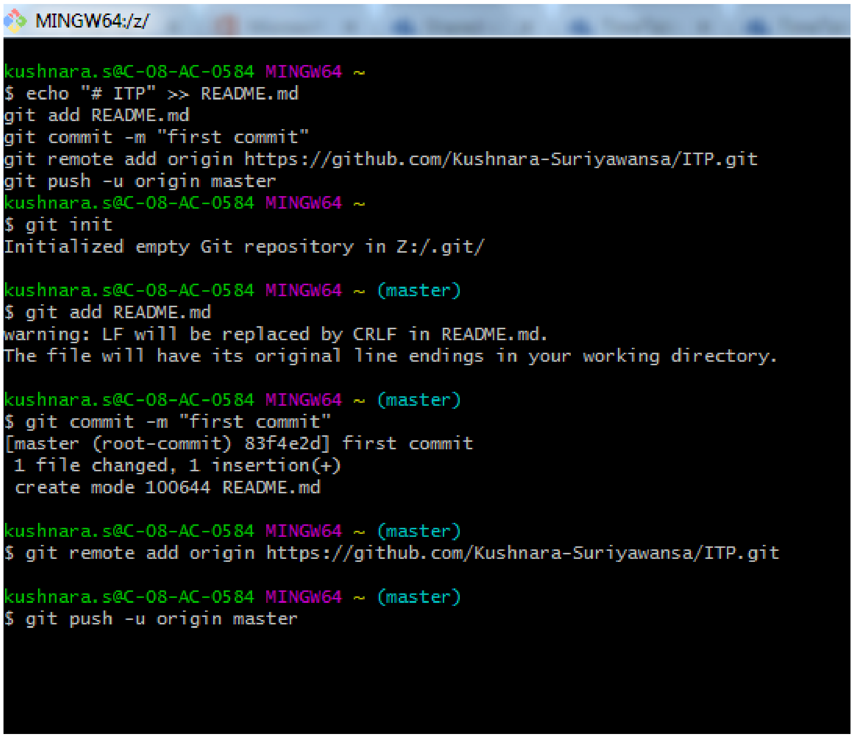
...or create a new repository on the command line

```
echo "# ITP" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/Kushnara-Suriyawansa/ITP.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/Kushnara-Suriyawansa/ITP.git
git push -u origin master
```

- *Open Git Bash*
- *Paste and execute the commands in Git Bash*

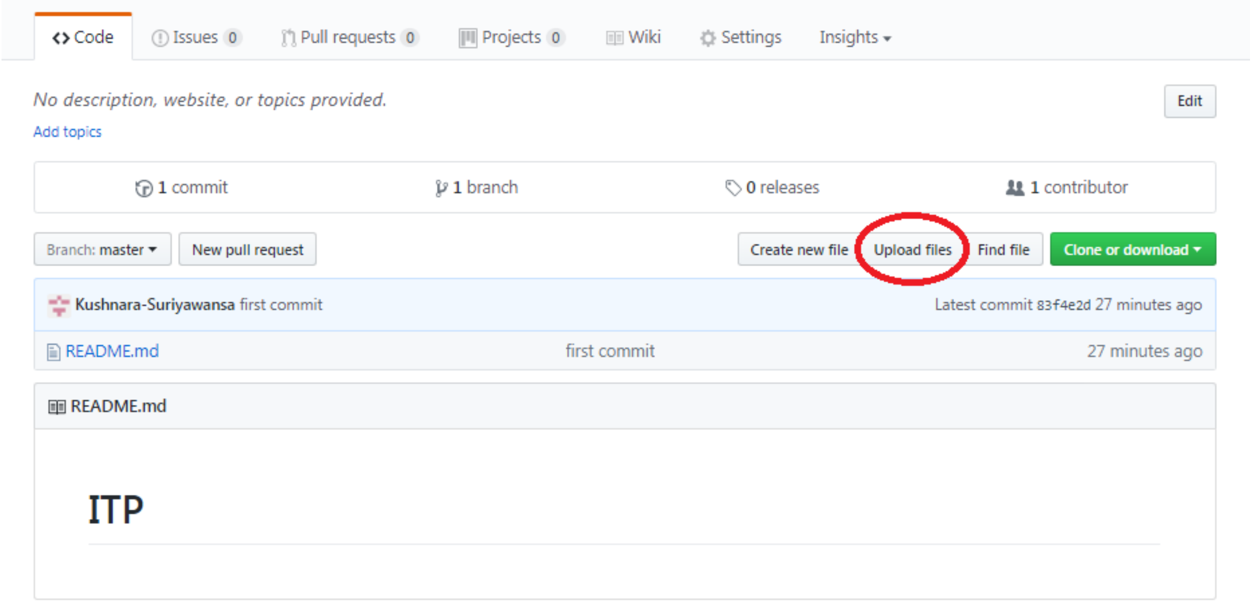


```
kushnara.s@C-08-AC-0584 MINGW64 ~
$ echo "# ITP" >> README.md
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/Kushnara-Suriyawansa/ITP.git
git push -u origin master
kushnara.s@C-08-AC-0584 MINGW64 ~
$ git init
Initialized empty Git repository in Z:/.git/

kushnara.s@C-08-AC-0584 MINGW64 ~ (master)
$ git add README.md
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory.

kushnara.s@C-08-AC-0584 MINGW64 ~ (master)
$ git commit -m "first commit"
[master (root-commit) 83f4e2d] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

kushnara.s@C-08-AC-0584 MINGW64 ~ (master)
$ git remote add origin https://github.com/Kushnara-Suriyawansa/ITP.git

kushnara.s@C-08-AC-0584 MINGW64 ~ (master)
$ git push -u origin master
```

- Give the username and password used in GitHub account.
- Refresh the GitHub in browser.

Now you will be able to see the committed README.md file in your repository.

04. Upload files to the repository



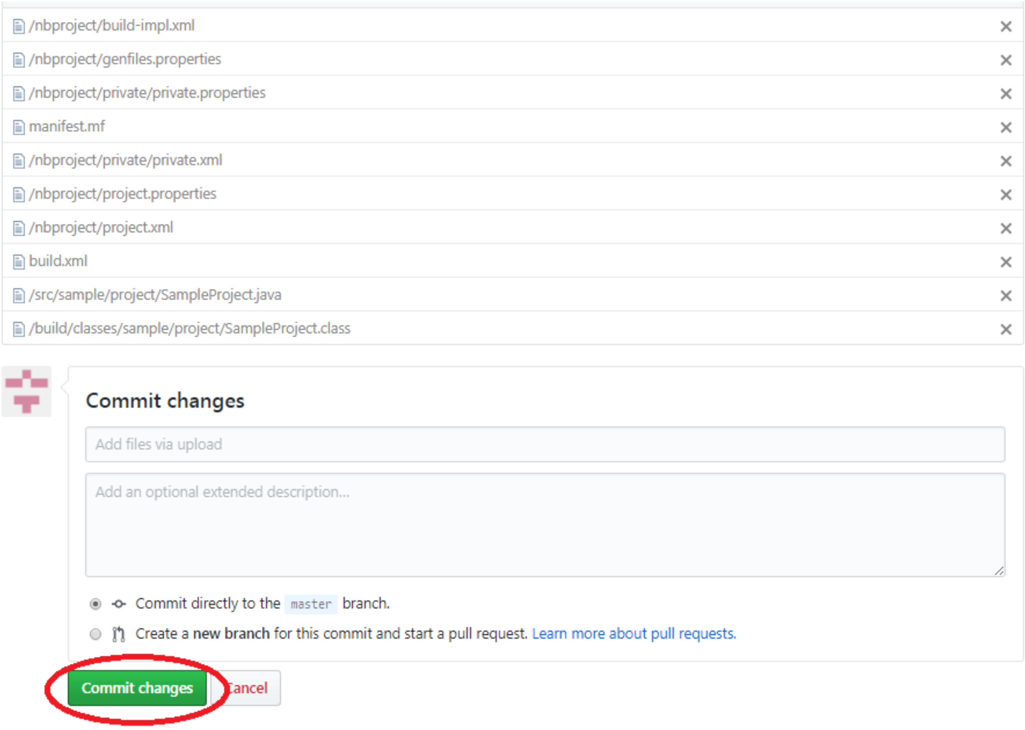
No description, website, or topics provided.

Kushnara-Suriyawansa first commit		Latest commit 83f4e2d 27 minutes ago
README.md	first commit	27 minutes ago

README.md

ITP

- Select Upload files option and add the files/folders of the given sample project.
- Commit changes to the repository



```
/nbproject/build-impl.xml
/nbproject/genfiles.properties
/nbproject/private/private.properties
manifest.mf
/nbproject/private/private.xml
/nbproject/project.properties
/nbproject/project.xml
build.xml
/src/sample/project/SampleProject.java
/build/classes/sample/project/SampleProject.class
```

Commit changes

Add files via upload

Add an optional extended description...

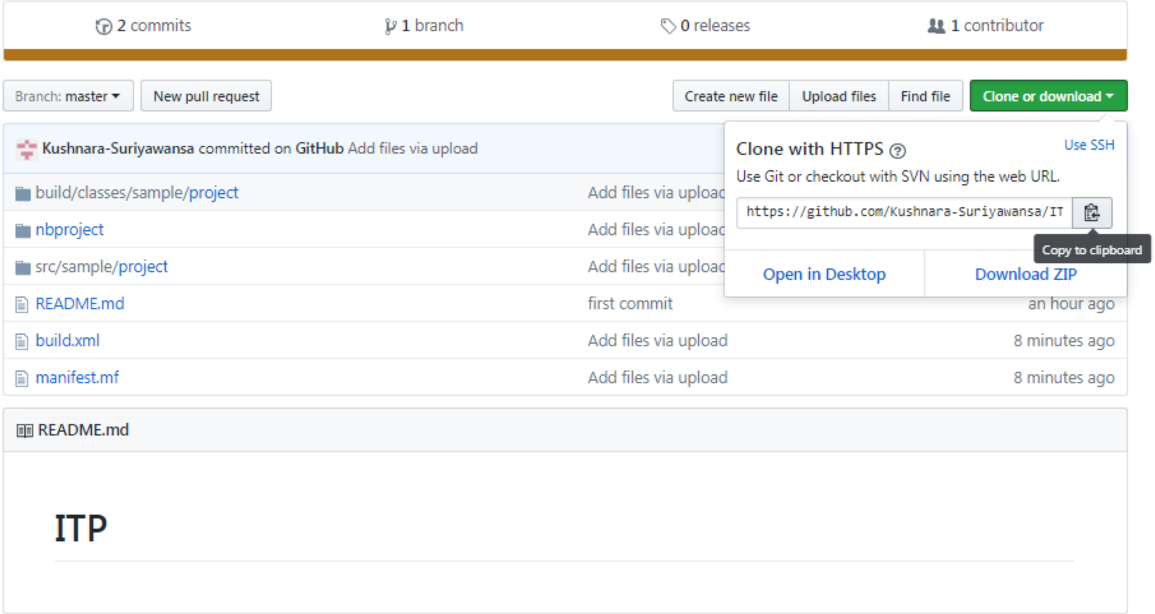
● Commit directly to the `master` branch.

○ Create a **new branch** for this commit and start a pull request. Learn more about pull requests.

Now you will be able to see the committed files/folders *in your repository.*

05. Clone the uploaded project to your desktop.

- Select Clone or Download option and copy the URL.



Kushnara-Suriyawansa committed on GitHub Add files via upload

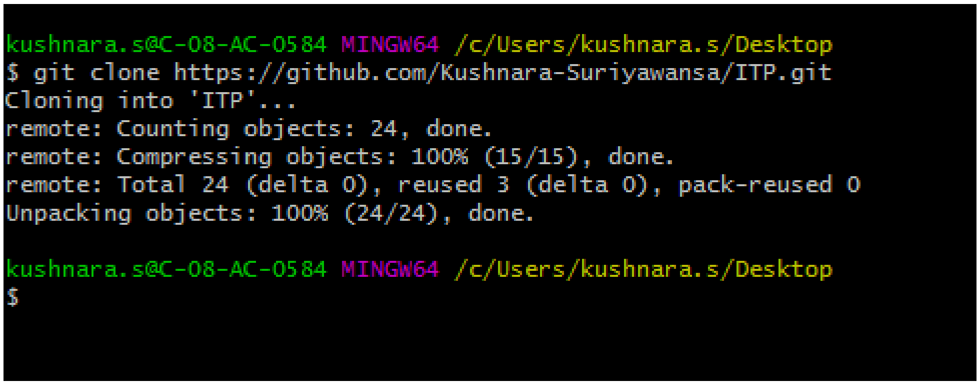
Use Git or checkout with SVN using the web URL.

https://github.com/Kushnara-Suriyawansa/IT

build/classes/sample/project	Add files via upload	
nbproject	Add files via upload	
src/sample/project	Add files via upload	
README.md	first commit	an hour ago
build.xml	Add files via upload	8 minutes ago
manifest.mf	Add files via upload	8 minutes ago

- Execute **git clone <URL>** command in Git Bash. (Open the GitBash from the Desktop)

  Note: ***git clone*** *command is used to clone/copy a repository to current directory of Git Bash. You have to provide the path/URL of the repository with this command.*
  Example: ***git clone https://github.com/username/ITP.git***



```
kushnara.s@C-08-AC-0584 MINGW64 /c/Users/kushnara.s/Desktop
$ git clone https://github.com/Kushnara-Suriyawansa/ITP.git
Cloning into 'ITP'...
remote: Counting objects: 24, done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 24 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (24/24), done.

kushnara.s@C-08-AC-0584 MINGW64 /c/Users/kushnara.s/Desktop
$
```

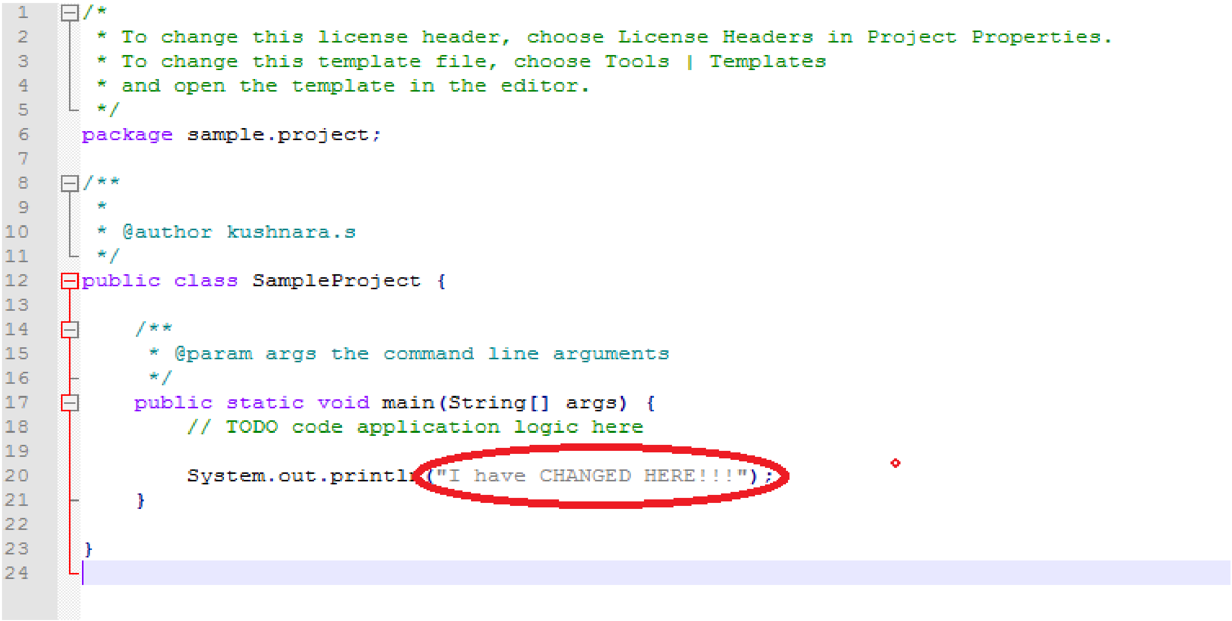
Now there will be a folder from the name of the repository on your desktop. This folder will have all files of the project that are uploaded to the GitHub repository. By default this project is initialized as a git project and you will be able to see the hidden .git folder. Thus, now you are able to execute git commands using the git bash for this cloned project.

06. Executing git commands.

Note : To execute the git commands to the project, open Git Bash inside the project folder. (Go inside the folder-> right click -> Git Bash here)

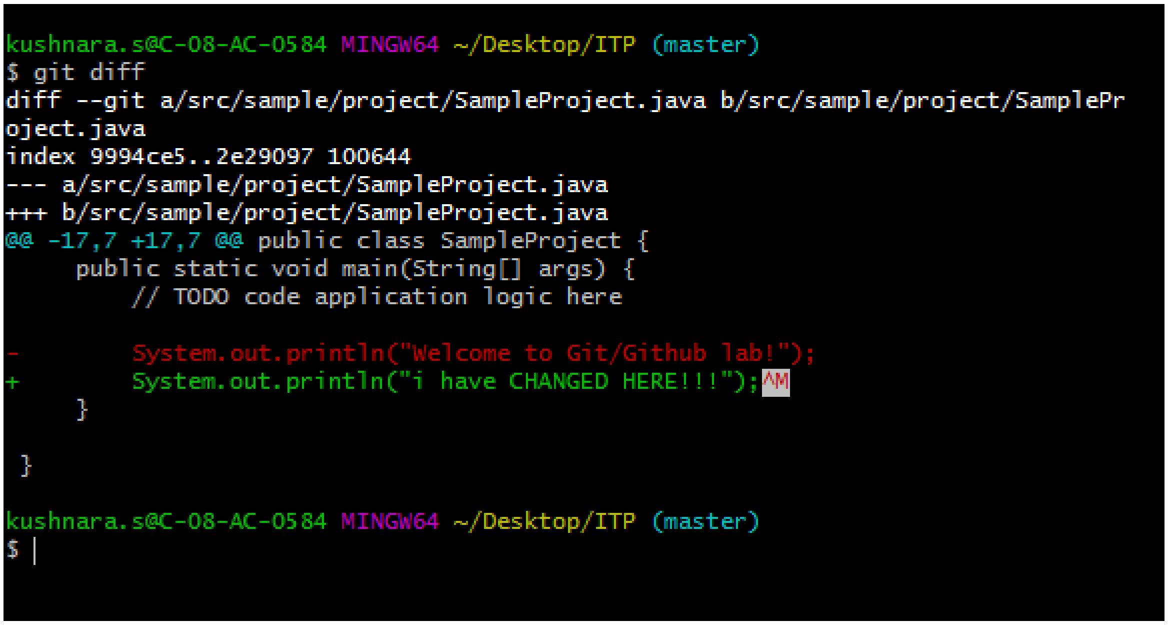
- **git diff :** Gives the changes (not yet saved/added) that are made.

  □ Open SampleProject .java file in the cloned project folder (in \src\sample\project path) and change the code as follows.



```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package sample.project;

/**
 *
 * @author kushnara.s
 */
public class SampleProject {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here

        System.out.println("I have CHANGED HERE!!!");
    }

}
```

□ Save the file and execute **git diff** command in Git Bash.



```
kushnara.s@C-08-AC-0584 MINGW64 ~/Desktop/ITP (master)
$ git diff
diff --git a/src/sample/project/SampleProject.java b/src/sample/project/SamplePr
oject.java
index 9994ce5..2e29097 100644
--- a/src/sample/project/SampleProject.java
+++ b/src/sample/project/SampleProject.java
@@ -17,7 +17,7 @@ public class SampleProject {
     public static void main(String[] args) {
         // TODO code application logic here

-        System.out.println("Welcome to Git/Github lab!");
+        System.out.println("i have CHANGED HERE!!!");^M
     }

 }

kushnara.s@C-08-AC-0584 MINGW64 ~/Desktop/ITP (master)
$ |
```

Notice how the changes are displayed in Git Bash.

- **git add <file name> :** Add/Save changes that are made to the given file.

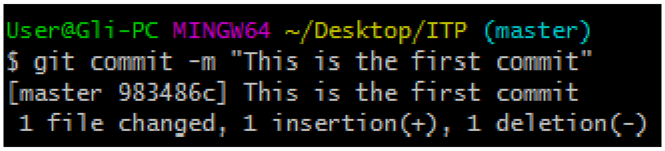
□ Execute **git add SampleProject .java** command in Git Bash.

Note: Give the path to the file that has to be added.
Ex: ***git add src/sample/project/SampleProject.java***

□ Again execute **git diff** command in Git Bash.
Note: The changes you have made are not shown by the git diff commands, as now they are added/saved.

- **git commit –m "<Commit Message>" :** Record the changes as a commit to local repository. Commit can be thought of as a version of the project.

□ Execute **git commit -m "This is the first commit"** command in Git Bash.



```
User@Gli-PC MINGW64 ~/Desktop/ITP (master)
$ git commit -m "This is the first commit"
[master 983486c] This is the first commit
 1 file changed, 1 insertion(+), 1 deletion(-)
```

- **git status :** Gives the status of the git repository.

    - □ Execute **git status** command in Git Bash.

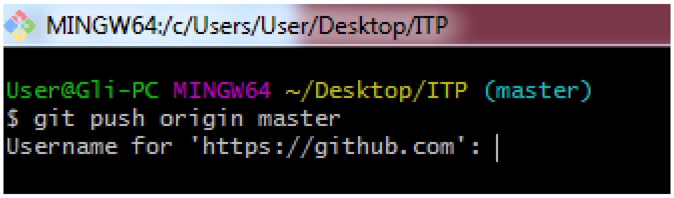
- **git log :** Gives the history of all commits. A log record contains commit hash, author, date and message of a commit.

    - □ Execute **git log** command in Git Bash.

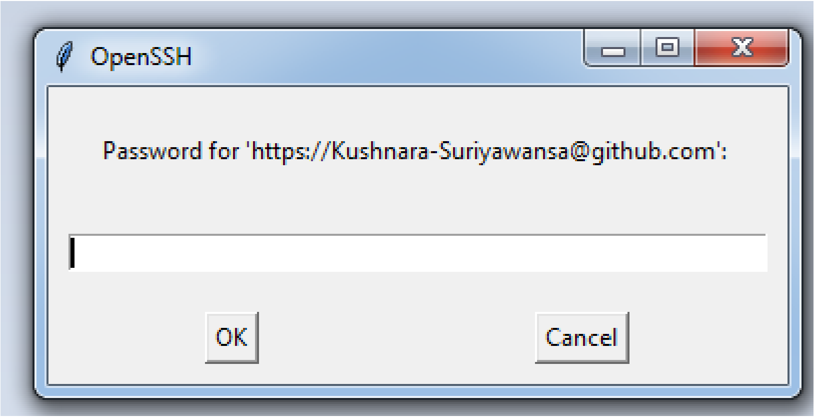
- **git push origin master:** Send the changes to master repository.

    - □ Execute **git push origin master** command in Git Bash.

      Make sure that you have executed git commit command before executing git push origin master, so that there is a change in the version of git repository from the origin master repository. Once you execute this command you will be asked to provide the user name and password of your GitHub account.



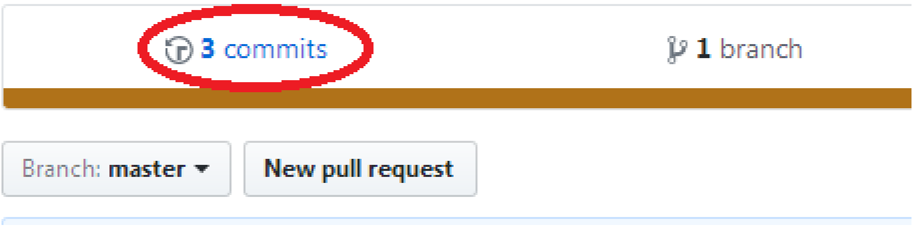
```
User@Gli-PC MINGW64 ~/Desktop/ITP (master)
$ git push origin master
Username for 'https://github.com': |
```

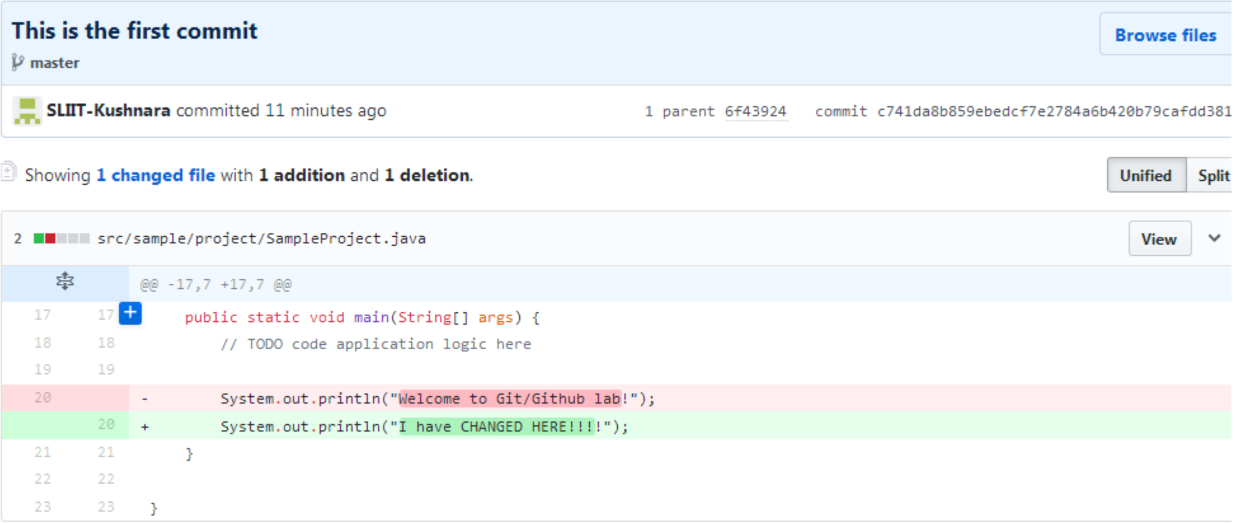


Password for 'https://Kushnara-Suriyawansa@github.com':

Now current version of your project repository is committed to the master repository that is in GitHub.

You will be able to see the commit request in the master project repository of GitHub.





This is the first commit

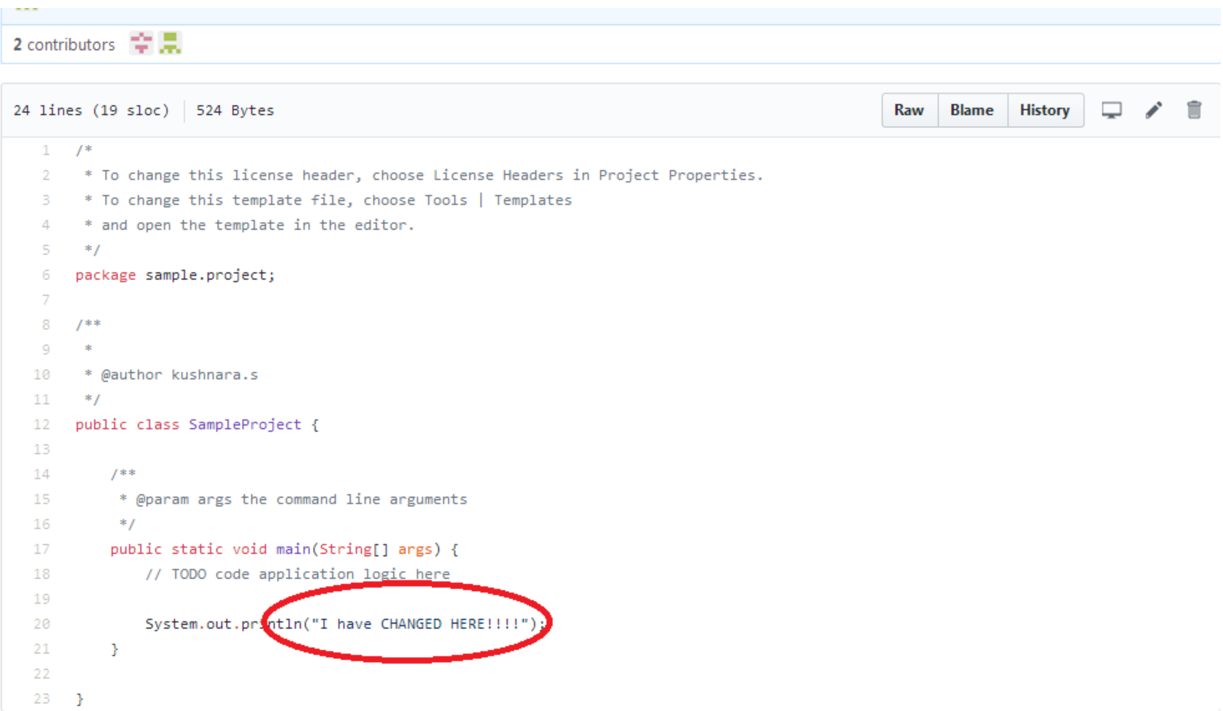
master

SLIIT-Kushnara committed 11 minutes ago   1 parent 6f43924   commit c741da8b859ebedcf7e2784a6b420b79cafdd381

Showing **1 changed file** with **1 addition** and **1 deletion**.

```
2 src/sample/project/SampleProject.java

@@ -17,7 +17,7 @@
      public static void main(String[] args) {
          // TODO code application logic here

-         System.out.println("Welcome to Git/Github lab!");
+         System.out.println("I have CHANGED HERE!!!!");
      }

  }
```

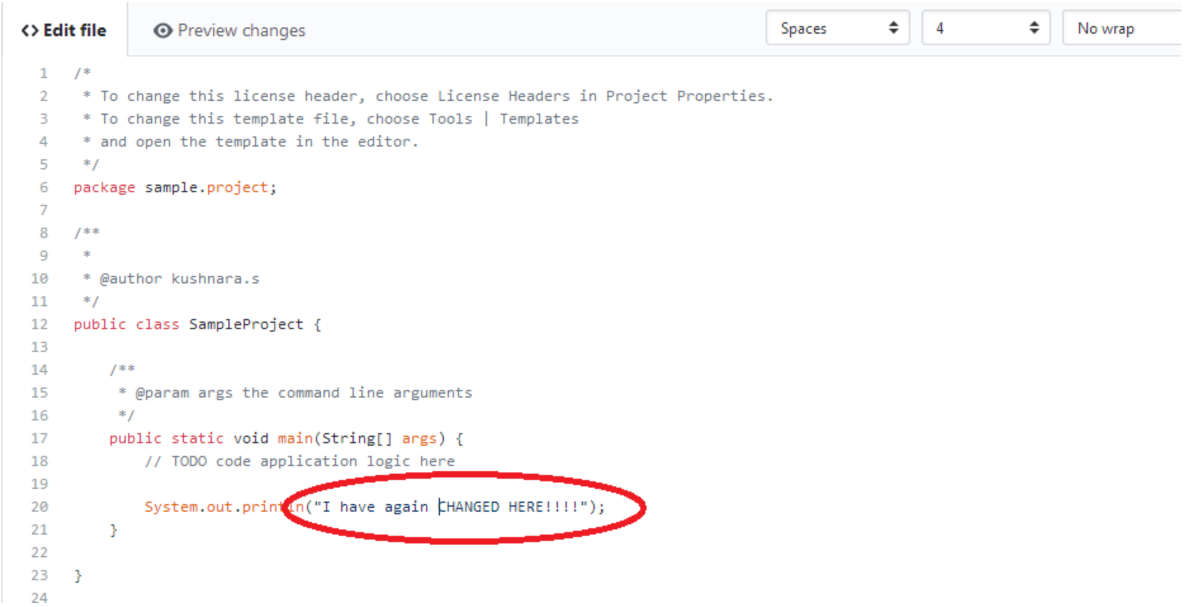
Note that the SampleProject.java file of the master project will now have the change you have made in local project.



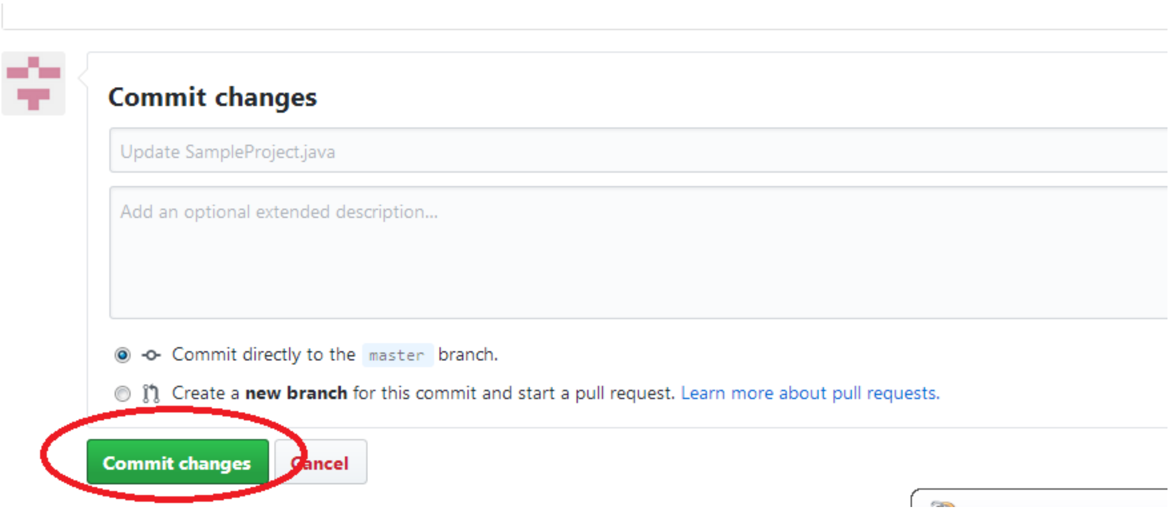
```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package sample.project;

/**
 *
 * @author kushnara.s
 */
public class SampleProject {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here

        System.out.println("I have CHANGED HERE!!!!");
    }

}
```

- **git pull :** Get the most recent version of the project from the master repository (in GitHub) to your computer.

    - Edit the SampleProject.java file in GitHub and commit the change.



```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package sample.project;

/**
 *
 * @author kushnara.s
 */
public class SampleProject {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here

        System.out.println("I have again CHANGED HERE!!!!");
    }

}

```



Commit changes

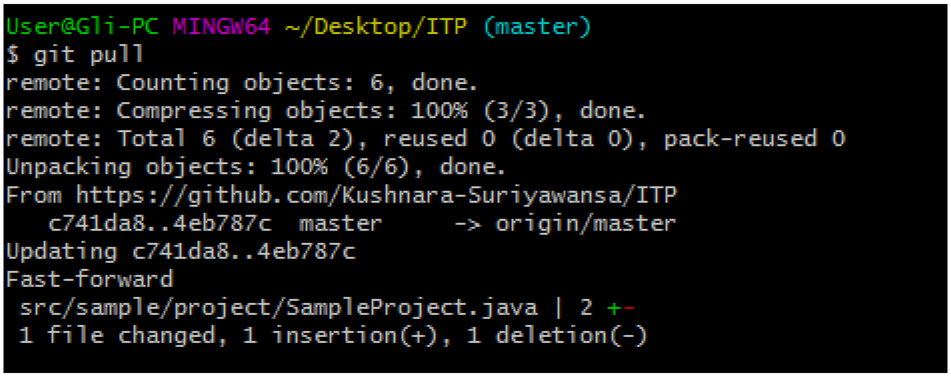
Update SampleProject.java

Add an optional extended description...

● Commit directly to the master branch.

○ Create a **new branch** for this commit and start a pull request. Learn more about pull requests.

□ Execute **git pull** command in Git Bash



```
User@Gli-PC MINGW64 ~/Desktop/ITP (master)
$ git pull
remote: Counting objects: 6, done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 6 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), done.
From https://github.com/Kushnara-Suriyawansa/ITP
   c741da8..4eb787c  master     -> origin/master
Updating c741da8..4eb787c
Fast-forward
 src/sample/project/SampleProject.java | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

Now changes you have made in master repository will be available in the project of your computer.

- **git reset --hard <commit hash>:** Revert files and undo the changes done in the commit. Here the commit hash is the hash id given for the commit which you can get from the output of git log command. ***Try this command later.***

A branch in Git is simply a lightweight movable pointer to one of these commits. The default branch name in Git is master.

- **git branch <name> :** Creates a new branch in the repository.
  - Execute **git branch newBranch** command in Git Bash.
- **git branch :** Display current branches in the repository.
  - Execute **git branch** command in Git Bash.
- **git checkout <branch name> :** Move to the branch given in the command.
  - Execute **git checkout newBranch** command in Git Bash.
  - Execute **git branch** command again in Git Bash.
- **git merge <branch name> :** Merge current branch with the branch given in the command.
  - Change the code in SapmleProject.java file, add the change and commit.
  - Execute **git merge master** command in Git Bash.
- **git branch –D <branch name> :** Delete the branch.
  - Execute **git branch –D newBranch** command in Git Bash.