Teacher’s Name:

Ms. Samia Kiran

Submitted By:

01-136221-020
Muhammad Talha

Department of Computer Science,

Bahria University, Islamabad.

Table of Contents

How to Set Up Git for the First Time

Introduction

Setting up Git for the first time involves configuring your user information, setting up SSH keys for secure communication with GitHub, and ensuring your local Git environment is correctly configured to push changes to a remote repository. This guide will walk you through the necessary steps, including managing files and folders in your repository.

1. Configure Git User Information

When you set up Git, it is important to configure your user information. This information is used to label the commits you make to the repository.

Steps:

1. Open Terminal or Command Prompt:

  - For Windows, open Command Prompt or Git Bash.

  - For macOS/Linux, open the Terminal.

2. Set Global Username and Email:

```
git config --global user.name "Your Name"

git config --global user.email "your.email@example.com"
```

Why: The `--global` flag ensures that these settings apply to all repositories on your machine. Git uses this information to identify the author of each commit.

2. Initialize a New Git Repository

To start using Git in a new project, you need to initialize a Git repository in your project directory.

Steps:

1. Navigate to Your Project Directory:

```
cd path/to/your/project
```

2. Initialize Git Repository

```
git init
```

Why: This creates a new `. git` directory in your project, which Git uses to track changes.

3. Add Files and Folders to the Repository

Once your repository is initialized, you need to add files and folders to it. In this case, you have a folder of images and a `model.ipynb` file.

Steps:

1. Add Files:

```
git add.
```

Why: The `git add.` command stages all changes in your working directory for the next commit. This includes the `model.ipynb` file and any other files or folders like your images.

2. Verify Changes:

```
git status
```

Why: This command shows the status of changes in your working directory and staging area.

4. Commit Changes

Committing saves the staged changes to the local repository.

Steps:

1. Commit Files:

```
git commit -m "Initial commit with model.ipynb and images folder"
```

Why: The commit message should briefly describe the changes made. This helps track the history of your project.

5. Set Up Remote Repository

To push your commits to a remote repository like GitHub, you need to add a remote URL.

Steps:

1. Add Remote Repository:

```
git remote add origin git@github.com:username/repository.git
```

Why: This links your local repository with the remote one on GitHub, allowing you to push and pull changes.

6. Set Up SSH Key for Authentication

SSH keys are used for secure communication between your machine and GitHub.

Steps:

1. Generate SSH Key:

```
ssh-keygen -t rsa -b 4096 -C "your.email@example.com"
```

Follow the prompts to save the key and optionally add a passphrase.

2. Add SSH Key to GitHub:

- Copy the contents of your public key (typically found at `~/.ssh/id_rsa.pub`).

- Go to GitHub Settings > SSH and GPG keys > New SSH key, and paste your public key there.

3. Test SSH Connection:

```
ssh -T git@github.com
```

Why: The SSH key allows GitHub to authenticate you without requiring a password every time. Testing the connection ensures that the key was added correctly.

7. Push Changes to Remote Repository

With everything set up, you can now push your local changes to the remote repository.

Steps:

1. Push Changes:

```
git push -u origin main
```

Why: The `-u` flag sets the upstream branch for the local `main` branch, allowing you to use `git push` without specifying the remote branch in the future.

8. Troubleshooting Common Issues

- Permission Denied (publickey):

  - Ensure your SSH key is correctly added to GitHub.

  - Verify the SSH key is being used by running `ssh-add -l`.

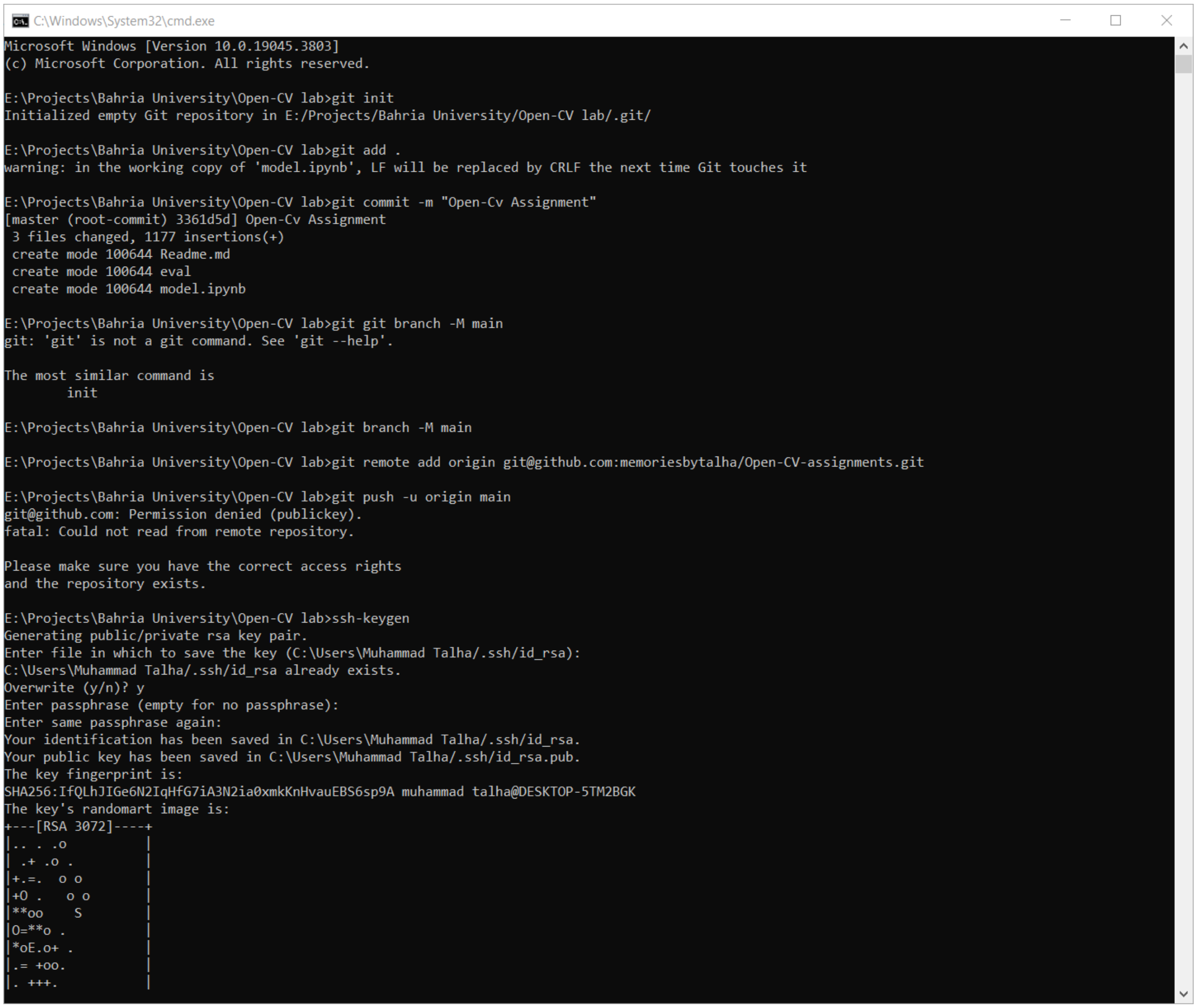
- `chmod` Not Recognized:

  - On Windows, use Git Bash or PowerShell to set file permissions if needed.

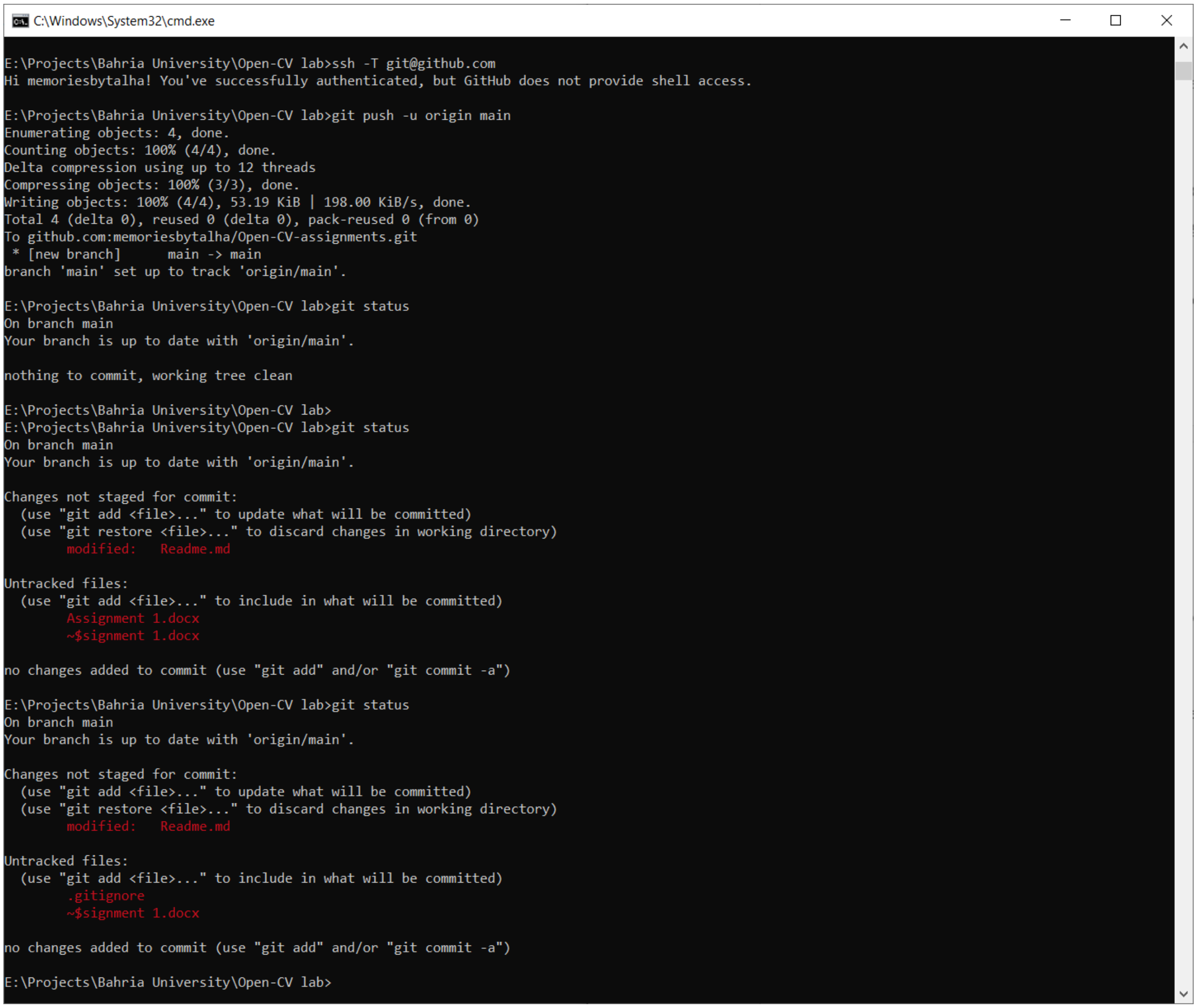
Conclusion

By following these steps, you’ve set up Git to manage your project and securely communicate with GitHub. You have also added important files and folders, including a Jupyter notebook and images.

Screenshots



```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

E:\Projects\Bahria University\Open-CV lab>git init
Initialized empty Git repository in E:/Projects/Bahria University/Open-CV lab/.git/

E:\Projects\Bahria University\Open-CV lab>git add .
warning: in the working copy of 'model.ipynb', LF will be replaced by CRLF the next time Git touches it

E:\Projects\Bahria University\Open-CV lab>git commit -m "Open-Cv Assignment"
[master (root-commit) 3361d5d] Open-Cv Assignment
 3 files changed, 1177 insertions(+)
 create mode 100644 Readme.md
 create mode 100644 eval
 create mode 100644 model.ipynb

E:\Projects\Bahria University\Open-CV lab>git git branch -M main
git: 'git' is not a git command. See 'git --help'.

The most similar command is
        init

E:\Projects\Bahria University\Open-CV lab>git branch -M main

E:\Projects\Bahria University\Open-CV lab>git remote add origin git@github.com:memoriesbytalha/Open-CV-assignments.git

E:\Projects\Bahria University\Open-CV lab>git push -u origin main
git@github.com: Permission denied (publickey).
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

E:\Projects\Bahria University\Open-CV lab>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\Muhammad Talha/.ssh/id_rsa):
C:\Users\Muhammad Talha/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\Muhammad Talha/.ssh/id_rsa.
Your public key has been saved in C:\Users\Muhammad Talha/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:IfQLhJIGe6N2IqHfG7iA3N2ia0xmkKnHvauEBS6sp9A muhammad talha@DESKTOP-5TM2BGK
The key's randomart image is:
+---[RSA 3072]----+
|.. . .o          |
| .+ .o .         |
|+.=.  o o        |
|+O .   o o       |
|**oo    S        |
|O=**o .          |
|*oE.o+ .         |
|.= +oo.          |
|. +++.           |
```



```
E:\Projects\Bahria University\Open-CV lab>ssh -T git@github.com
Hi memoriesbytalha! You've successfully authenticated, but GitHub does not provide shell access.

E:\Projects\Bahria University\Open-CV lab>git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 53.19 KiB | 198.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:memoriesbytalha/Open-CV-assignments.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

E:\Projects\Bahria University\Open-CV lab>git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

E:\Projects\Bahria University\Open-CV lab>
E:\Projects\Bahria University\Open-CV lab>git status
On branch main
Your branch is up to date with 'origin/main'.

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

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Assignment 1.docx
        ~$signment 1.docx

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

E:\Projects\Bahria University\Open-CV lab>git status
On branch main
Your branch is up to date with 'origin/main'.

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

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore
        ~$signment 1.docx

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

E:\Projects\Bahria University\Open-CV lab>
```