WEEK 2

TUTORIAL 4 - GIT & GITHUB

CHEAT SHEET

LINNAEUS
UNIVERSITY
SUMMER 2024

CREATING, CLONING & PUSHING GIT REPOSITORIES

In this tutorial, we will learn how to create a separate Git repository, obtain the repository URL, clone it to your local machine, add a file to it, and commit the changes. Then we will create a local Git repository, add a folder and file to it, commit the changes, create a repository on GitHub, and push the local repository to GitHub.

1 SET UP GIT AND GITHUB

- Install Git on your computer by downloading it from the official website and following the installation instructions.
- Create an account on GitHub if you don't have one already. Sign in to your GitHub account.

2 CREATE A SEPARATE GIT REPOSITORY

- Go to your GitHub profile and click on the "+" sign at the top right corner.
- Select "New repository" to create a new repository.
- Provide a name for your repository, choose the desired settings, and create the repository.

3 OBTAIN THE SEPARATE REPOSITORY URL

- On the repository page, click on the "Code" tab in the top right.
- Copy the repository's URL (ends with .git) (example https://github.com/ElenaSeroka/test.git)
- This URL will be used to clone the repository to your local machine.

4 CLONE THE SEPARATE REPOSITORY

- Create a folder where you will keep this repository. Go into the directory.
- Open your terminal or command prompt in the directory.
- Run ```git clone ```, replacing with the copied URL.
- Press Enter to clone the repository to your local machine.

5 ADD A FILE TO THE SEPARATE REPOSITORY

- Change into the cloned repository's directory using ```cd repository_name```, where "repository_name" is the name of the cloned repository.
- Create a new file within the repository.
- Open the file using a text editor and add your desired content.

6 STAGE AND COMMIT CHANGES IN THE SEPARATE REPOSITORY

- Run ```git status``` to check the status of your repository and see the untracked file.
- Add the file to the staging area by running ```git add filename.ext``` Replace "filename.ext" with the actual name of your file.
- Use ```git status``` to verify that the changes are staged and ready for commit.
- Commit the changes by running ```git commit -m "Added a new file"```
- The "Added a new file" is a customizable message. You can type anything you want!
- If there is an issue, and Github wants you to verify your identity, do so! Follow their prompts.
- Before you send up you changes, you ALWAYS take down the current status of the repository as it is in the cloud.
- To do that you type ```git pull```
- Type ```git push``` to upload your changes.
- Check your GitHub!

7 CREATE A LOCAL GIT REPOSITORY

- Create a new directory.
- Open your terminal or command prompt in that directory.
- Type ```git init``` in the terminal to initialize a new Git repository in that directory.

8 ADD AND COMMIT CHANGES IN THE LOCAL REPOSITORY

- Create a new file within the folder. Add something to that file if it is a text file.
- Add the files to your git by typing ```git add .``` ("add ." means add all files)
- Use ```git status``` to verify that the changes are staged and ready for commit.
- Commit the changes by running ```git commit -m "your message here"```.

9 CREATE A REPOSITORY ON GITHUB

- Go to the GitHub website and sign in to your account.
- Click on the "+" sign at the top right corner and select "New repository".
- Provide a name for your repository, choose the desired settings, and create the repository

10 CONNECT LOCAL REPOSITORY TO REMOTE REPOSITORY

- On the GitHub repository page, copy the repository's URL (ends with .git).
- Back in your terminal, run the command ```git remote add origin ```, replacing with the copied URL.
- Verify that the remote repository is added by running ```git remote -v```.

11 PUSH LOCAL REPOSITORY TO GITHUB

- Run ```git push -u origin master``` to push the local repository to the remote GitHub repository.
- Provide your GitHub credentials when prompted.
- Wait for the push to complete, and you will see the changes reflected in your GitHub repository.

GIT BASH / TERMINAL TEXT COMMAND CHEATSHEET

ACTION	TYPE	COMMAND
CREATE	FILE	TOUCH FILENAME.EXTENSION
DESTROY	FILE	RM FILENAME.EXTENSION
CREATE AND WRITE TO A FILE	FILE	• **CAT > FILENAME.EXTENSION** After running this command, the terminal will wait for you to input text. This text will be redirected and written to filename.extension. • **Type text you want to be saved in filename.extension.** • **Ctrl+D** Press Ctrl+D to save the content to filename.extension.
RENAME	FILE	MV OLDNAME.EXTENSION NEWNAME.EXTENSION
MOVE	FILE	MV FILENAME.EXTENSION DIRECTORY
CREATE	DIRECTORY	MKDIR DIRECTORYNAME
DESTROY	DIRECTORY	RM -R DIRECTORYNAME
RENAME	DIRECTORY	MV OLDNAME NEWNAME
MOVE	DIRECTORY	MV NAME DIRECTORY