

Working with Github – quick start guide

Introduction

Github is a website which facilitates group working on software development projects. For the purposes of this course it can be used:

- (1) For the course tutor to provide you with sample files, and
- (2) For you to save your own files from the virtual machines, so that you can access them later.

To get files from Github to your virtual machine, or to put files onto Github we use a source code management application called git. This application is pre-installed on your virtual machine but can be downloaded and installed on your personal computer from <https://git-scm.com/>

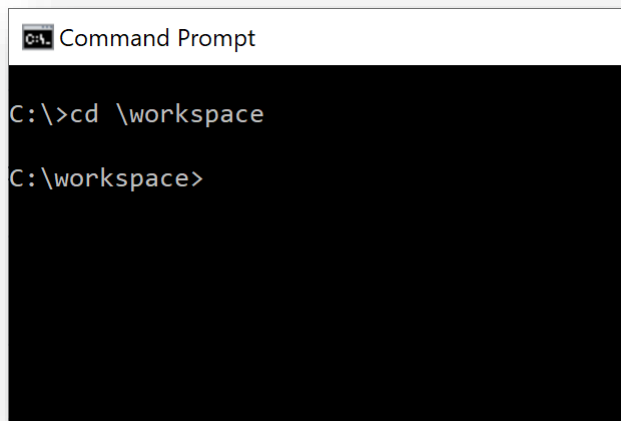
In Github we work with repositories – a repository is a complete set of files (which could be 1 or more projects).

1 – preparing your computer

We suggest that you create a folder called workspace at the top level of the C drive, and put all the files from your course tutor, or that you create, under this folder.

The easiest way to use git is to open a command prompt (click on the search icon, type in `cmd` and press enter).

To navigate to this folder in a command prompt, enter the command: `cd \workspace` and press enter.



```
Command Prompt
C:\>cd \workspace
C:\workspace>
```

Before you use git for the first time, you should configure it by entering the following 2 commands. (Replace the section in quotes with your own details)

```
git config --global user.name "your name"
```

```
git config --global user.email "your email"
```

```
Command Prompt

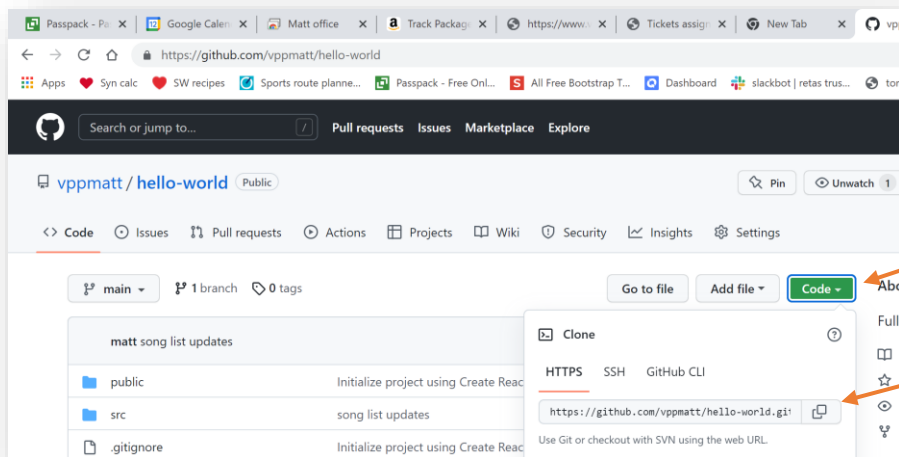
C:\workspace>git config --global user.name "Matt Thornfield"

C:\workspace>git config --global user.email "matt@multicode.co.uk"

C:\workspace>
```

2 – Copy files to your computer

To copy files from Github onto your computer, start by visiting the Github web page for the repository, click on the green code button, and then click on the copy icon in the box which appears, to copy the git URL.



Then on your computer, in the command prompt, navigate to the workspace folder and enter the command

```
git clone <paste here the url you copied>
```

and press enter. To paste into a windows command prompt, just press the right mouse button.

```
Command Prompt

C:\>cd \workspace

C:\workspace>git clone https://github.com/vppmatt/hello-world.git
Cloning into 'hello-world'...
remote: Enumerating objects: 39, done.
remote: Counting objects: 100% (39/39), done.
remote: Compressing objects: 100% (32/32), done.
remote: Total 39 (delta 6), reused 39 (delta 6), pack-reused 0
Unpacking objects: 100% (39/39), done.

C:\workspace>
```

If the course tutor updates the files and you wish to update your local copy, ensure you are in the folder for the relevant project, and enter the command:

`git pull.`

3 – Using a github account

If you wish to put your own files onto Github (which you will need to do if you want to save your files from the virtual machine to access later), you will need an account on the github.com website. If you already have an account, you can use this. Alternatively, your course tutor may provide you with an account you can use, or you can create a free account on the github.com website yourself.

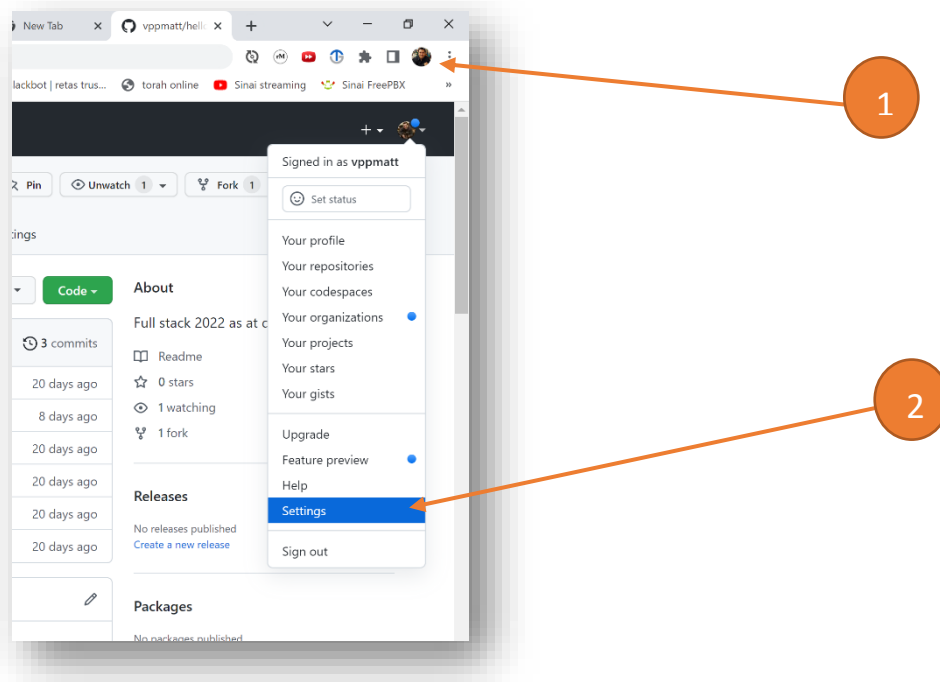
It is strongly recommended that you sign into your Github account using a browser on the virtual machine.

4 – Get a personal access token

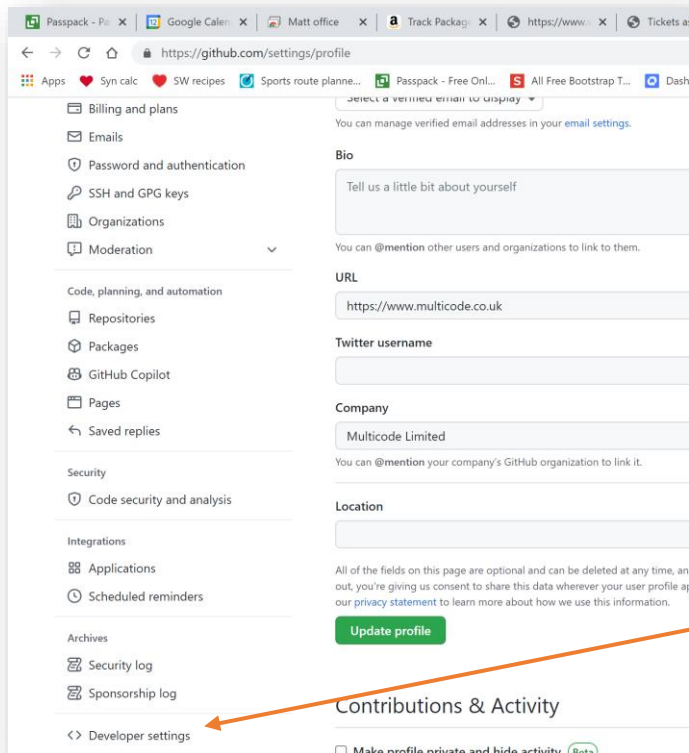
Once you have access to an account on Github.com and you are signed into the account on your browser, you will need to create a personal access token. You will use this instead of a password to copy files from your computer to your Github account.

To create the token:

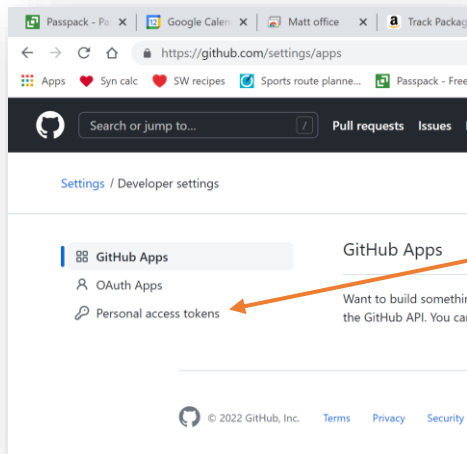
Visit the Github website, and click on the profile icon at the top right of the page. Then choose “Settings” from the dropdown menu.



Scroll down the page and on the left hand side click on “Developer settings”



Now click on “Personal access tokens”



Click on the “generate new token” button. On the next screen, give the token a name. We suggest you choose “no expiration” from the dropdown box. Then tick EVERY top level box before clicking on the green “Generate Token” button.

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of your password to authenticate to the GitHub API over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

Example token

What's this token for?

Expiration *

No expiration The token will never expire!

GitHub strongly recommends that you set an expiration date for your token to help keep your account secure. [Learn more](#)

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repos:status	Access commit status
<input checked="" type="checkbox"/> repos:deployment	Access deployment status
<input type="checkbox"/> write:repo_hook	Write repository hooks
<input checked="" type="checkbox"/> read:repo_hook	Read repository hooks
<input checked="" type="checkbox"/> admin:org_hook	Full control of organization hooks
<input checked="" type="checkbox"/> gist	Create gists
<input checked="" type="checkbox"/> notifications	Access notifications
<input checked="" type="checkbox"/> user	Update ALL user data
<input checked="" type="checkbox"/> read:user	Read ALL user profile data
<input checked="" type="checkbox"/> user:email	Access user email addresses (read-only)
<input checked="" type="checkbox"/> user:follow	Follow and unfollow users
<input checked="" type="checkbox"/> delete:repo	Delete repositories
<input checked="" type="checkbox"/> write:discussion	Read and write team discussions
<input checked="" type="checkbox"/> read:discussion	Read team discussions
<input checked="" type="checkbox"/> admin:enterprise	Full control of enterprises
<input checked="" type="checkbox"/> manage:runners:enterprise	Manage enterprise runners and runner-groups
<input checked="" type="checkbox"/> manage:billing:enterprise	Read and write enterprise billing data
<input checked="" type="checkbox"/> read:enterprise	Read enterprise profile data
<input checked="" type="checkbox"/> project	Full control of projects
<input checked="" type="checkbox"/> read:project	Read access of projects
<input checked="" type="checkbox"/> admin:gpg_key	Full control of public user GPG keys
<input checked="" type="checkbox"/> write:gpg_key	Write public user GPG keys
<input checked="" type="checkbox"/> read:gpg_key	Read public user GPG keys

Generate token **Cancel**

Numbered callouts: 5 points to the 'Example token' field; 6 points to the 'Expiration' section; 7 points to the 'repo' scope; 8 points to the 'Generate token' button.

On the next page, you will be shown your token – you must copy this token (click on the copy icon) and save it somewhere (such as in a notepad file on the desktop of the virtual machine).

If you lose the token, you will need to create a new one to replace it.

Personal access tokens **Generate new token** **Revoke all**

Tokens you have generated that can be used to access the GitHub API.

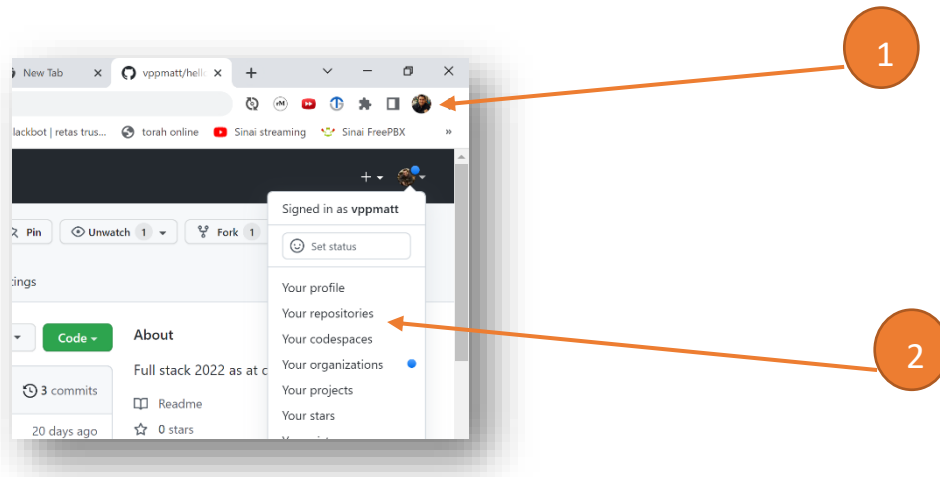
Make sure to copy your personal access token now. You won't be able to see it again!

✓ ghp_sYtbhKksykBcFxzUUhigditds8ihbC1n3eJk **Delete**

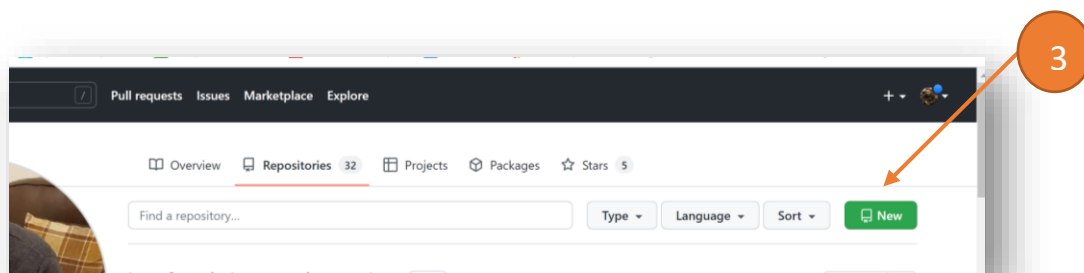
Numbered callout: 9 points to the copy icon next to the token.

5 – Create a Github repository

For each project that you wish to copy to Github you first need to create a repository. Go to the repositories page (click on the profile link and then repositories)



Click on the green New button



Give the repository a name – don't put any spaces in the name. This name should ideally match the folder name for the project on your computer (it doesn't have to match but it is better if it does!)

Then click on the green “create repository” button

Owner ⁺ vppmatt / Repository name ⁺ example-project ✓

Great repository names are **example-project is available**. Need inspiration? How about [laughing-octo-potato](#)?

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

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

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)
.gitignore template: None ▼

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)
License: None ▼

ⓘ You are creating a public repository in your personal account.

Create repository

IMPORTANT - DO NOT NAVIGATE AWAY FROM THE NEXT PAGE!

6 – Create a local git repository

The next step is to make your code into a git repository. Within your project folder, enter the following commands to create the local git repository and put all your files into it

```
git init
```

```
git add .
```

```
git commit -m "first commit"
```

```

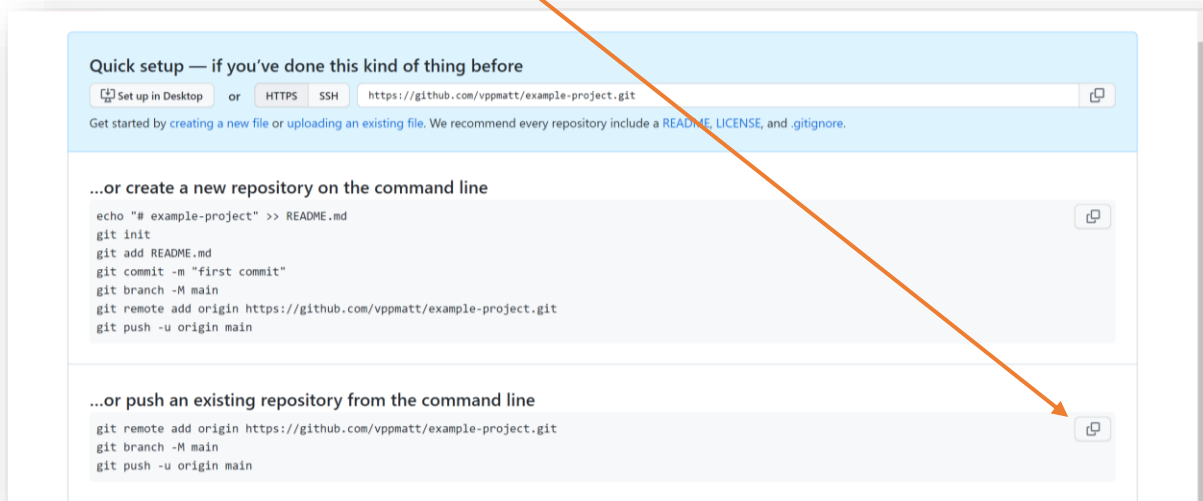
C:\workspace\exmaple-project>git init
Initialized empty Git repository in C:/workspace/exmaple-project/.git/

C:\workspace\exmaple-project>git add .

C:\workspace\exmaple-project>git commit -m "first commit"
[master (root-commit) 7bdcd5e] first commit
1 file changed, 1 insertion(+)
create mode 100644 test.txt

C:\workspace\exmaple-project>
```

Now copy the middle section from the Github website, and paste this into the command prompt.



```
Command Prompt

C:\workspace\exmaple-project>git init
Initialized empty Git repository in C:/workspace/exmaple-project/.git/

C:\workspace\exmaple-project>git add .

C:\workspace\exmaple-project>git commit -m "first commit"
[master (root-commit) 7bdc5e] first commit
1 file changed, 1 insertion(+)
create mode 100644 test.txt

C:\workspace\exmaple-project>git remote add origin https://github.com/vppmatt/example-project.git

C:\workspace\exmaple-project>git branch -M main

C:\workspace\exmaple-project>git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 220 bytes | 73.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/vppmatt/example-project.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.

C:\workspace\exmaple-project>
```

The first time that you do this, you will be asked for your username and password. This will appear in a pop up window. DO NOT ENTER YOUR DETAILS HERE – just click on the cancel button to close that window.

You will then be asked to enter your username and password in the command prompt.

Enter your github username, but instead of the password, enter the developer access token. Remember that you can right click to paste in the command prompt. Note that as the token is a password, it will not appear on screen when you paste it.

7 – Store your token

A hidden folder will have been created called `.git`, and within this folder is a file called `CONFIG`. Open this file in a text editor such as notepad.

Contained within this file is the URL to your repository. Edit this line, inserting your Github username and the token you copied so that it is in the following format:

`https://username@token:github.com/...`



```
*config - Notepad
File Edit Format View Help
[core]
  repositoryformatversion = 0
  filemode = false
  bare = false
  logallrefupdates = true
  symlinks = false
  ignorecase = true
[remote "origin"]
  url = https://vppmatt@ghp_SyF9fsphFS2349FadfiaohF:github.com/vppmatt/example-project.git
  fetch = +refs/heads/*:refs/remotes/origin/*
[branch "main"]
  remote = origin
  merge = refs/heads/main
```

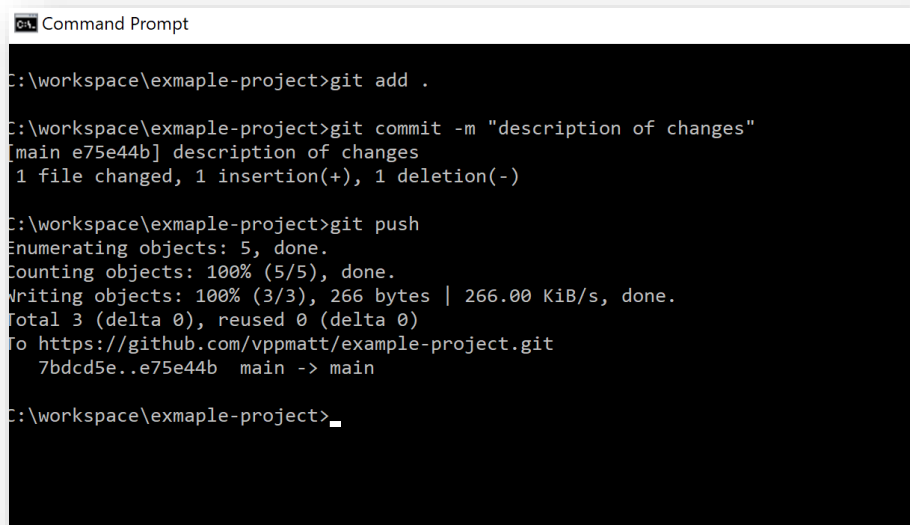
8 – Synchronize your changes

As you make changes to your code, you can synchronize these with the Github website by entering the following commands

`git add .`

`git commit -m "some message"`

`git push`



```
Command Prompt

C:\workspace\exmaple-project>git add .

C:\workspace\exmaple-project>git commit -m "description of changes"
[main e75e44b] description of changes
1 file changed, 1 insertion(+), 1 deletion(-)

C:\workspace\exmaple-project>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 266 bytes | 266.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/vppmatt/example-project.git
  7bdcd5e..e75e44b  main -> main

C:\workspace\exmaple-project>
```