

**ALL** Step-by-step instructions  
used in the DFCEP course  
**up to and including May 31**

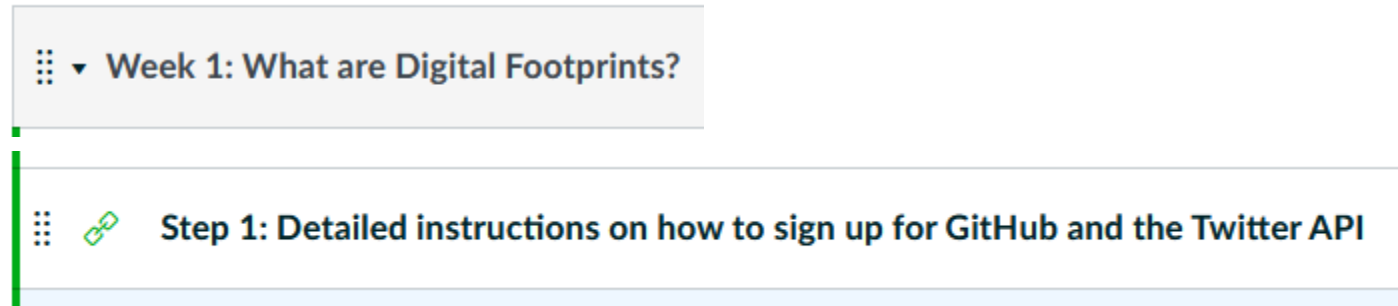
Dr. Ana Martinovici

# Purpose of this file

- As we progress through the course, I will give you more step-by-step instruction slides.
- Although detailed instructions help, there is a risk of confusion about what instructions are in which presentation.
- To reduce this risk, this file contains ALL step-by-step instructions. You can think of this file as the MAIN FILE.
- Keep in mind that this file will grow over time as more slides are added.
- The slides are in the same order as they were shared during the course. This means that if you want to see the most recent slides, you need to go to the end of the presentation.

# Set 1

You find the Canvas version under Modules ->



**Start**

# Step-by-step instructions

## Week 1

Dr. Ana Martinovici

# This file gives step-by-step instructions for:

- Signing up for a GitHub account
- Applying for access to the Twitter API

# How to sign up for a GitHub account

**Start** of section

# Sign up for a GitHub account

- Go to: <https://github.com/>
- Click “Sign up”
- Use your EUR email address to sign up

Join GitHub

## Create your account

Username \*

XXXXXXXXXXXXX ✓

Email address \*

XXXXX @eur.nl ✓

Password \*

..... ✓

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

Email preferences

☐ Send me occasional product updates, announcements, and offers.

# Select 'Student' and the programming experience that is applicable to you.

What kind of work do you do, mainly?

Software Engineer I write code	<b>Student</b> I go to school
Product Manager I write specs	UX & Design I draw interfaces
Data & Analytics I write queries	Marketing & Sales I look at charts
Teacher I educate people	Other I do my own thing

How much programming experience do you have?










<b>None</b> I don't program at all	<b>A little</b> I'm new to programming
<b>A moderate amount</b> I'm somewhat experienced	<b>A lot</b> I'm very experienced



Select 'Learn Git and GitHub', 'Host a project (repository)', and 'School work and student projects'.

What do you plan to use GitHub for?

(Select up to 3)

 Learn to code	 Learn Git and GitHub	 Host a project (repository)
 Create a website with GitHub Pages	 Collaborating with my team	 Find and contribute to open source
 School work and student projects	 Use the GitHub API	 Other

# Finalize the sign up process

- It is possible to fill in your interests, so GitHub can connect you with communities and projects.
- Please note that this is not mandatory.

**I am interested in:**

languages, frameworks, industries

We'll connect you with communities and projects that fit your interests.

For example: `3d` `font` `gis`

- To finalize setting up your account, click 'Complete setup'

GitHub will verify your email address. Open the email they send and click 'Verify email address'.



## Please verify your email address

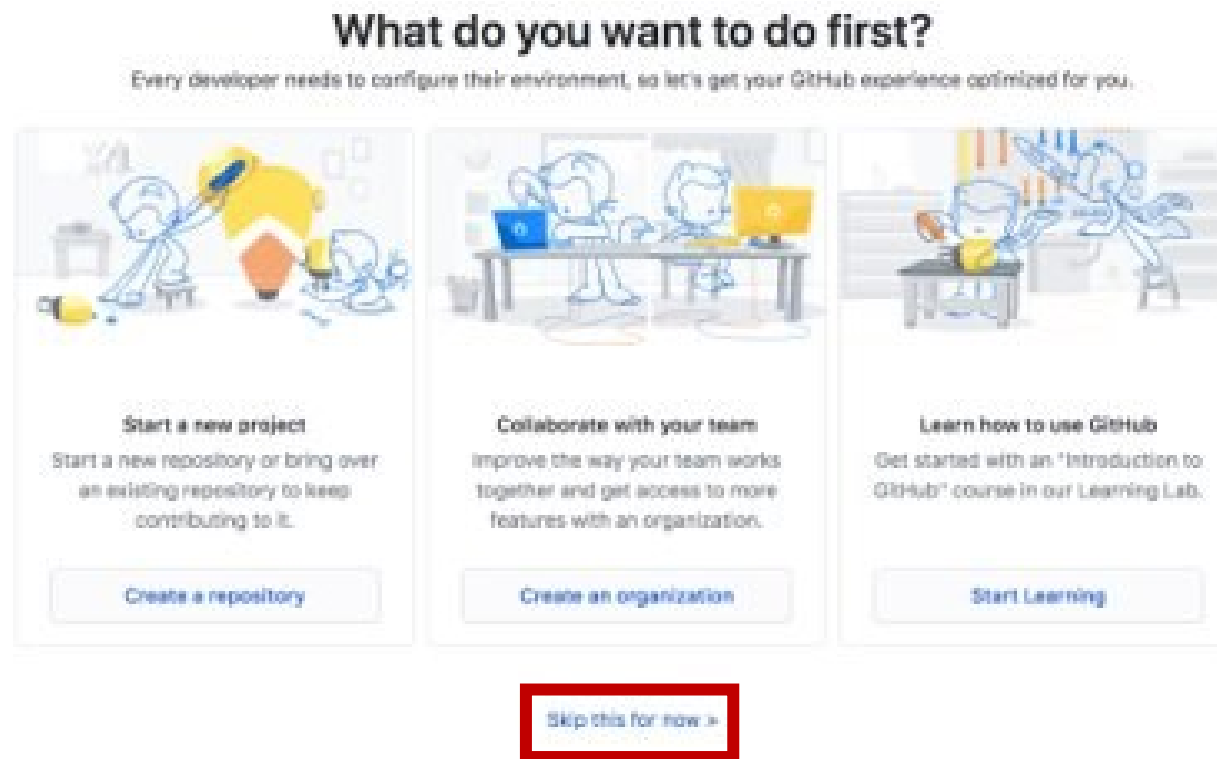
Before you can contribute on GitHub, we need you to verify your email address.

An email containing verification instructions was sent to `xxxxx @eur.nl`.

[Resend verification email](#)

[Change your email settings](#)

After you clicked 'Verify email address', you will see the following screen. You can click 'Skip this for now'.



# How to sign up for a GitHub account

**End** of section

# How to apply for access to the Twitter API

**Start** of section

Go to:

<https://developer.twitter.com/en/apply-for-access>

Click 'Apply for a  
developer account'

Get started with Twitter APIs and tools

# Apply for access

All new developers must apply for a developer account to access the Twitter developer platform. Once approved, you can begin to use our new Twitter API v2, or our v1.1 standard and premium APIs.

Apply for a developer account

Restricted use cases >

# Login to your Twitter account or create a new account by clicking on 'Sign up for Twitter'

If you sign up, then I recommend you use your EUR email address.



## Log in to Twitter

Phone, email, or username

Password


Log in


Forgot password? [Sign up for Twitter](#)





# Select 'Academic' and 'Student', then click 'Get Started'


**Which best describes you?**  
This is how you intend to use the Twitter developer platform


  
Professional


  
Hobbyist

  
**Academic**

  
Academic researcher

  
Teacher

  
**Student**

  
Something else  
(But related to academics)

**Standard application**  
Submit a standard application for access to the Twitter developer platform.

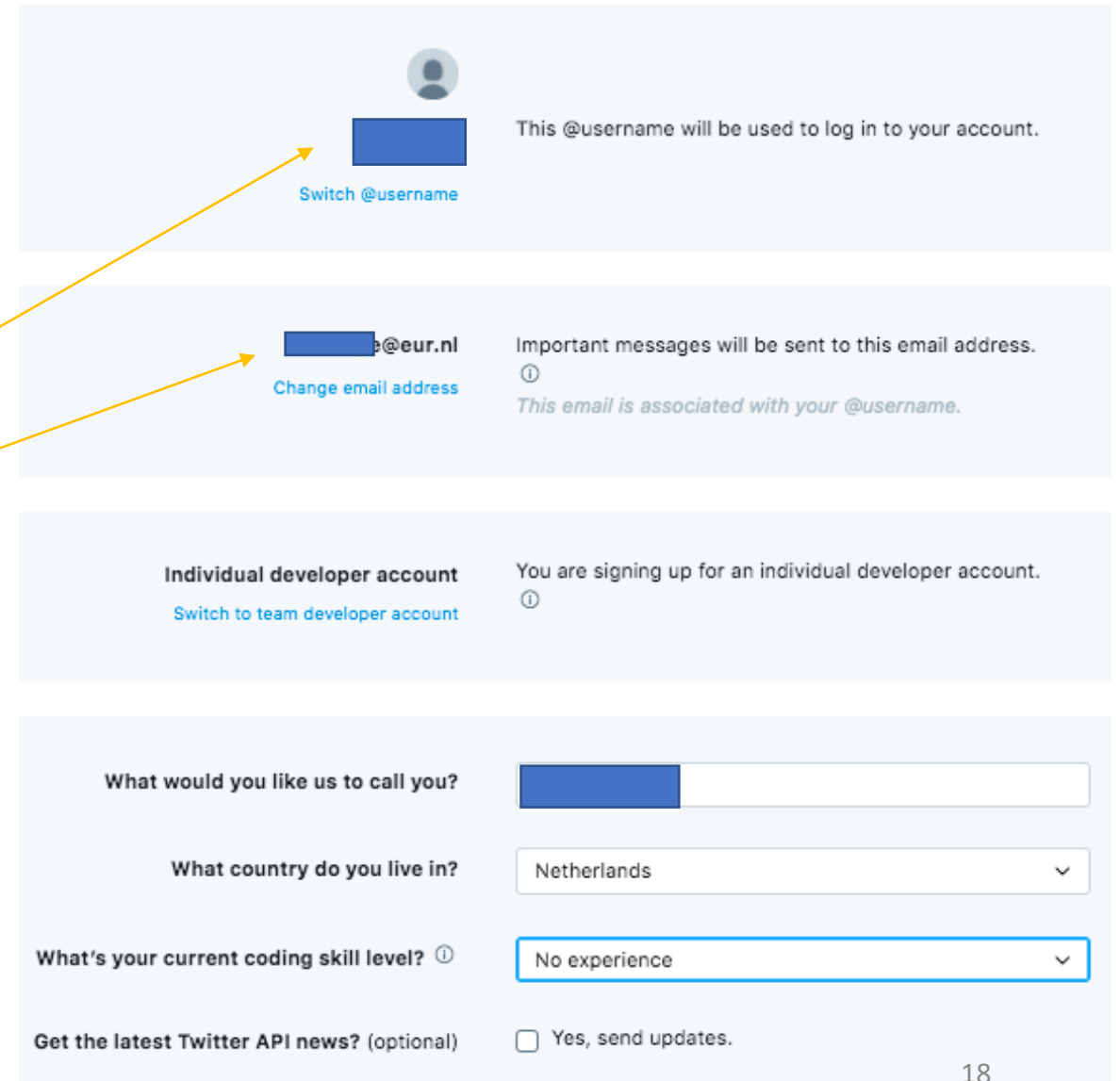
**Get started**

# Provide required info

To start your application, Twitter wants you to confirm your username and the associated email address/phone number. This is important as you will use them to log in to the Developer Platform.

**Note:** you should see your own information **here**.

Fill in with **your own details**.



The screenshot shows the Twitter Developer Platform account setup interface. It consists of four main sections:

- Username Selection:** A light blue box containing a profile icon placeholder, a text input field with a blue box, and a "Switch @username" link. To the right, it says "This @username will be used to log in to your account."
- Email Confirmation:** A light blue box containing a text input field with a blue box and "@eur.nl", a "Change email address" link, and explanatory text: "Important messages will be sent to this email address." and "This email is associated with your @username."
- Account Type:** A light blue box with the heading "Individual developer account", a "Switch to team developer account" link, and the text "You are signing up for an individual developer account."
- Personal Details:** A light blue box with four rows of form fields:
  - "What would you like us to call you?" with a text input field containing a blue box.
  - "What country do you live in?" with a dropdown menu showing "Netherlands".
  - "What's your current coding skill level?" with a dropdown menu showing "No experience".
  - "Get the latest Twitter API news? (optional)" with a checkbox labeled "Yes, send updates."

Yellow arrows from the text "you should see your own information here" point to the username and email input fields in the first two sections. A large yellow bracket on the left side of the bottom section groups the personal details fields.

# Provide required info

- In the next part of the application, Twitter wants to make sure that data is handled in accordance with their Developer Policies.
- The following slides provide examples of how you could answer the questions. You may use these examples if they are an accurate description of what you plan to use the app for.

# Example of how you will use the API and data

## How will you use the Twitter API or Twitter Data?

### In your words

In English, please describe how you plan to use Twitter data and/or APIs. The more detailed the response, the easier it is to review and approve.

I will collect Twitter data to complete an assignment for a university course that I am following. I will only get data and will not use the API to post information. Part of this course also covers GDPR and Ethical use of data, which will be taken into account when collecting data via the Twitter API.

# Example of how you will use the API and data

## The specifics

Please answer each of the following with as much detail and accuracy as possible. Failure to do so could result in delays to your access to Twitter developer platform or rejected applications.

Are you planning to analyze Twitter data?



Please describe how you will analyze Twitter data including any analysis of Tweets or Twitter users.

I will prepare graphs to show engagement (number of likes, retweets). I will only present results aggregated over multiple tweets or accounts.

100

# Example answer

Will your app use Tweet, Retweet, Like, Follow, or Direct Message functionality?

☒ No

Do you plan to display Tweets or aggregate data about Twitter content outside Twitter?

☒ Yes

**Please describe how and where Tweets and/or data about Twitter content will be displayed outside of Twitter.**

Tweets and aggregated data will be part of a final report that is required for this course. I will share the report only with the teacher of the course. Tweets or aggregated data about Twitter will not be publicly available outside of the course.

100

Now you can check your information again. If everything is correct click 'Next'.

If you read the Developer Agreement and the Twitter Developer Policy, you can click the box and submit your application.

- ☒ By clicking on the box, you indicate that you have read and agree to this [Developer Agreement](#) and the [Twitter Developer Policy](#), additionally as it relates to your display of any of the Content, the [Display Requirements](#); as it relates to your use and display of the Twitter Marks, the [Twitter Brand Assets and Guidelines](#); and as it relates to taking automated actions on your account, the [Automation Rules](#). These documents are available in hardcopy upon request to Twitter.

# Verify your email!

The review of your application begins only after you verified your email.



## Time to verify your email

We just sent an email verification to [redacted]@eur.nl.  
Once verified, your application review will begin.

Don't see it? Try checking your spam inbox or  
[have it resent.](#)

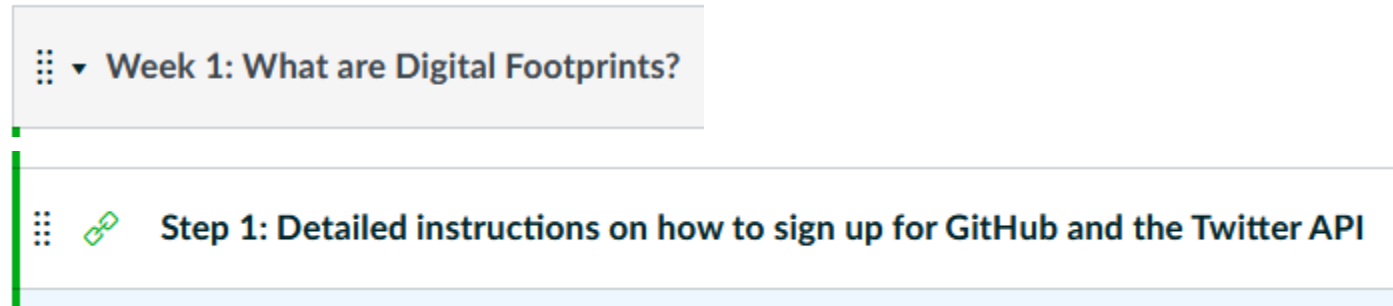


# How to apply for access to the Twitter API

**End** of section

# Set 1

You find the Canvas version under Modules ->

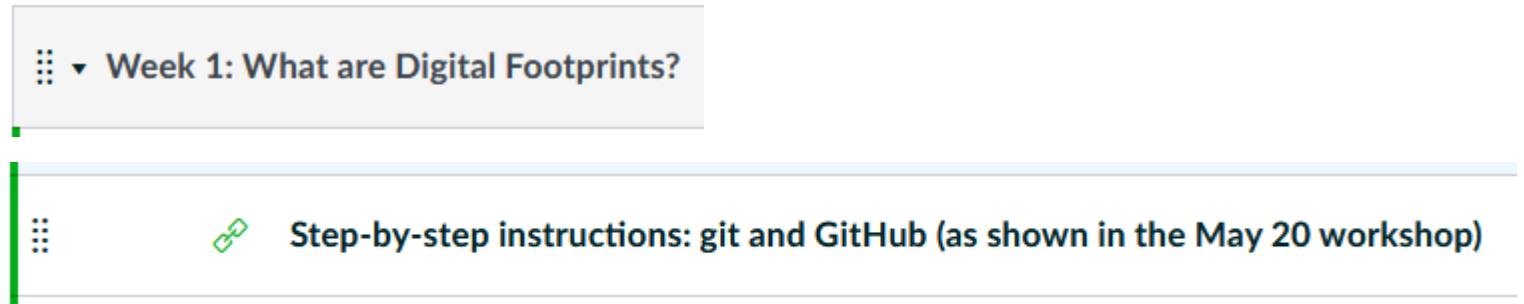


**End**



# Set 2

You find the Canvas version under Modules ->



**Start**

# Git and GitHub

## step-by-step instructions

Dr. Ana Martinovici

# Important note!

There are more ways of doing the same thing.

These slides show only one of these different ways.

If you have experience\* doing things differently, that's ok.

\*years of experience and an in-depth understanding of what you're doing

Info relevant for all the other  
instructions in this document

**Start** of section

# Vocabulary

- **repository (repo):** (*noun*) folder containing all tracked files as well as the version control history
- **remote:** (*noun*) the version of your repository that is stored on a remote server; for instance, on GitHub
- **clone:** (*verb*) to create a local copy of a remote repository on your personal computer
- **local:** (*noun*) the version of your repository that is stored on your personal computer



# Vocabulary

- **stage:** (*noun*) the staging area holds the files to be included in the next commit; (*verb*) to mark a file to be included in the next commit
- **commit:** (*noun*) a snapshot of changes made to the staged file(s); (*verb*) to save a snapshot of changes made to the staged file(s)
- **push:** (*verb*) to send commits from a local repository to a remote repository

# Vocabulary

- **fork:** (*noun*) a copy of another user's repository on GitHub; (*verb*) to copy a repository; for instance, from one user's GitHub account to your own
- **track:** (*noun*) a tracked file is one that is recognized by the Git repository
- **branch:** (*noun*) a parallel version of the files in a repository

# Vocabulary

- **merge:** (*verb*) to update files by incorporating the changes introduced in new commits
- **pull:** (*verb*) to retrieve commits from a remote repository and merge them into a local repository
- **pull request:** (*noun*) pull requests let you tell others about changes you've pushed to a branch

# Repositories you are using in this course

- **DFCP\_group\_project-YourTeamName**

- Each student group has a different repository.
- You are required to clone, commit & push 😊 - part of your group project
- This is where you share your code and data files for the group project

Note, May 20: You need a link to join this repo. I will share the link in Week 3.

# Repositories you are using in this course

- **DFCP-practice-YourGitHubUserName**

- Join by using this invite link: <https://classroom.github.com/a/cz1vSHYz>
- Each student has a different repository
- You may clone, commit & push 😊 to practice the Basics of GitHub
- This is where you can experiment working in an individual assignment repository, without being afraid of breaking things. So try out anything you want and see what happens.

# Repositories you are using in this course

- **dfcp\_test\_group\_project-YourTeamName**
  - Join by using this link: <https://classroom.github.com/g/6UNnGSh1>
  - Each student group has a different repository
  - You may clone, commit & push 😊 to practice the basics of GitHub
  - This is where you can experiment working in a group assignment repository, without being afraid of breaking things. So try out anything you want and see what happens.

# Repositories you are using in this course

- **DFCP\_public**

- [https://github.com/anamartinovici/DFCP\\_public](https://github.com/anamartinovici/DFCP_public)
- You have the right to clone this repository, but you cannot commit & push to it (if you try it will give you an error)
- This is where I will share code and code-related materials

# DFCP\_public

## What is this repository?

- This is a public repo where I am adding example code and other materials relevant for the Digital Footprints of Consumer Preferences course

## Why not post code and materials on Canvas?

- Sharing code via Canvas is like analyzing data in Excel – it is possible, many people do it, but it's not a good option.
- Canvas is ok for sharing recordings, pdf files, and other reading materials. Canvas is not meant to share code and has no functionalities for properly sharing code.

## What should I do with this repository?

- Check the section on “Recommended way of working with DFCP\_public”.



Info relevant for all the other  
instructions in this document

**End** of section

# Recommended way of working with repositories\*

**Start** of section

\*repositories you have the right to commit & push changes to

# Recommended way of working with repositories

## Notes:

- I'm referring to working with the repo for your group assignment, but these recommendations also apply to working with other repositories you can contribute to.
- By “work session” I mean: a time interval when you are working on your assignment, without interruptions. For example, if today you work on your assignment between 10.00 and 16.00, and you take a break between 12.00 and 14.00 -> the first work session is between 10.00 and 12.00 and the second session is between 14.00 and 16.00.
- This way of working is by no means the best or single option you have. I am recommending it because it helped me learn how to use version control on my own. If you have extensive experience with version control and other computing practices, you already have your own style – go with that. If version control is new to you and you don't fully understand what you're doing, then following the recommended way of working can help.

# Structure of a work session

1. Start by cloning the repo
2. Do the actual work
  - create / modify files in this repo (e.g., collect data, process data from the API, do the analysis, write the report).
3. Stage -> commit -> push changes to the remote
  - Write informative commit messages (i.e., messages that explain **why** you've changed the files).
4. Check that the remote shows the correct information
  - See section "Recommended way of working with DFCEP\_public", slide "The remote can change"
5. Delete the local repository from your laptop

As you gain experience using version control, you will stop doing step 4 and you will do step 5 less often.

# What does this structure achieve?

- Makes sure that you are able to submit your assignment via GitHub.
  - If you encounter issues with stage -> commit -> push, you have time to fix them before the deadline.
- Prevents losing work
  - If something happens to your laptop, you can still access all the data & code added to GitHub by using another device.
- Makes it easier to ask questions about your assignment
  - via GitHub you can add questions for specific files or lines of code, which is much better than sending “my code gives an error” via email.

# Recommended way of working with repositories\*

**End** of section

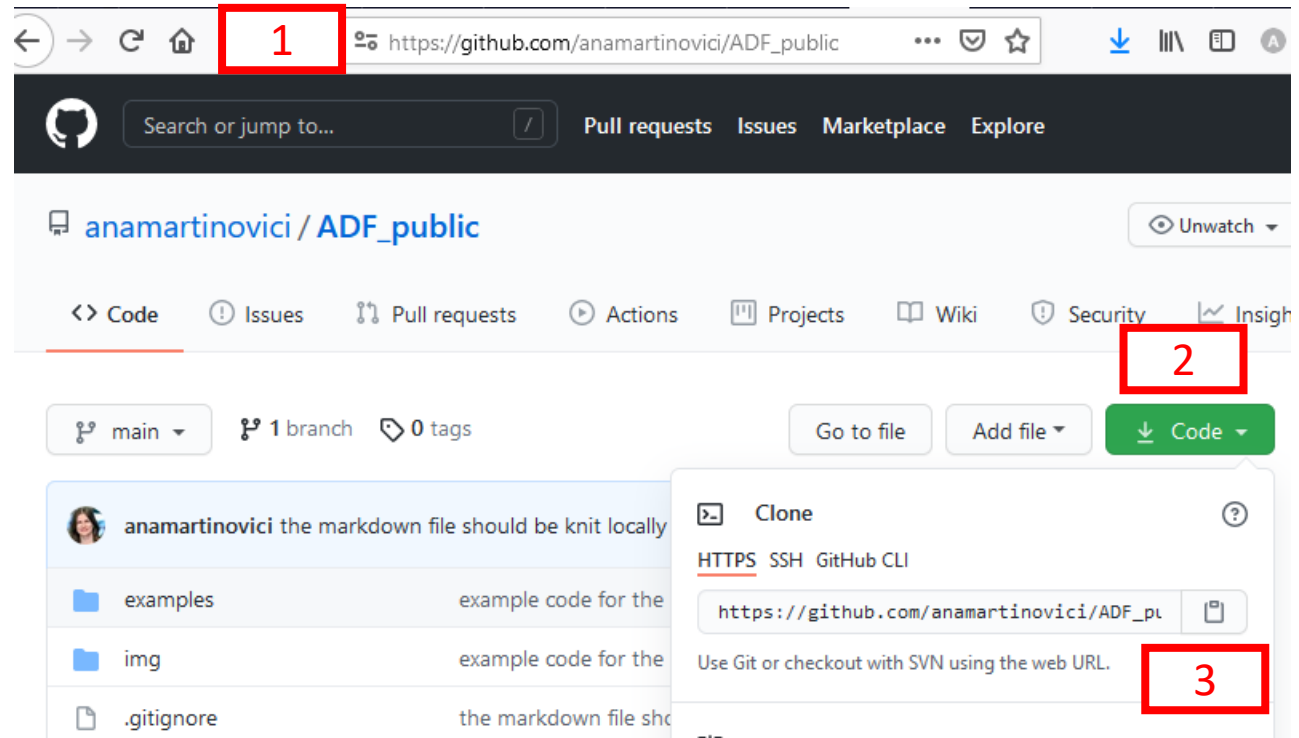
\*repositories you have the right to commit & push changes to

# How to clone a repo

**Start** of section

The example uses a repo named ADF\_public, but the steps are the same for other repositories you have access to

1. Go to the GitHub page of that repo  
[https://github.com/anamartinovici/ADF\\_public](https://github.com/anamartinovici/ADF_public)
2. Click “Code”
3. Copy [https://github.com/anamartinovici/ADF\\_public](https://github.com/anamartinovici/ADF_public)





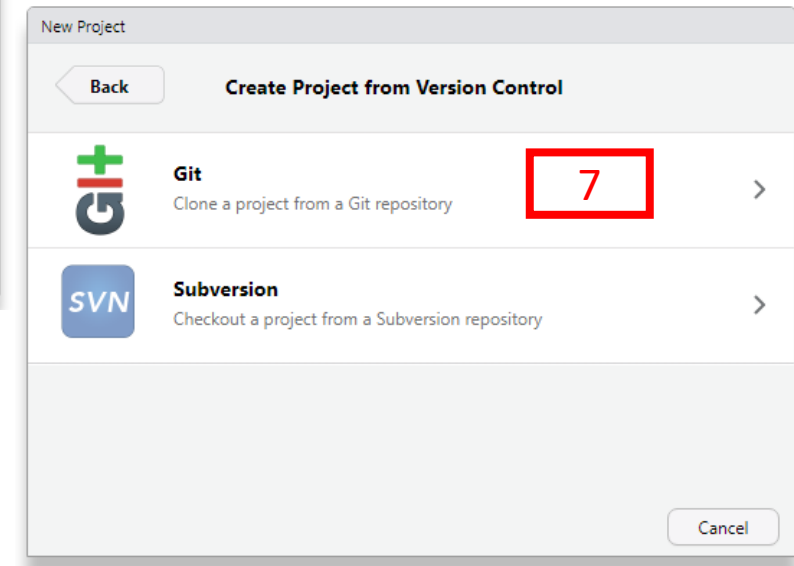
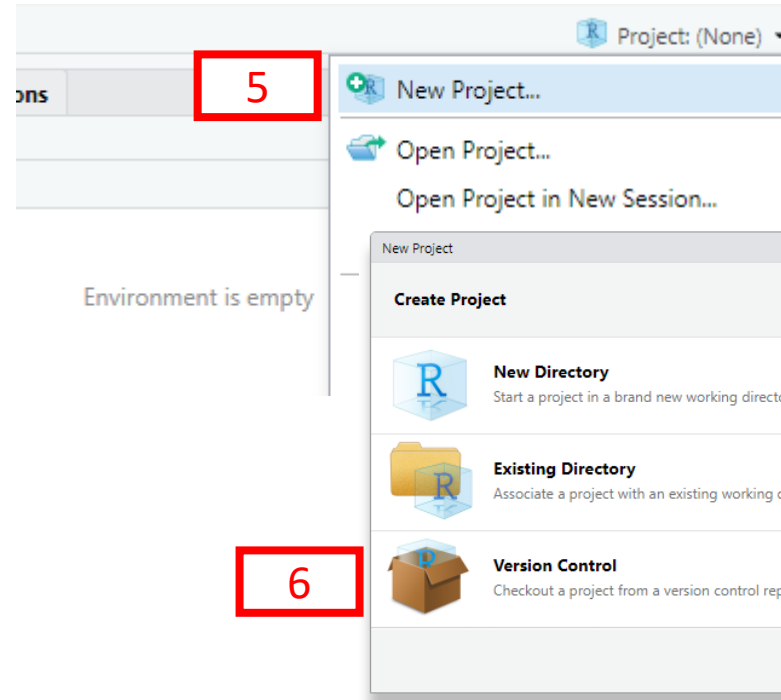
The example uses a repo named ADF\_public, but the steps are the same for other repositories you have access to

4. Open Rstudio

5. New project

6. Version control

7. Git

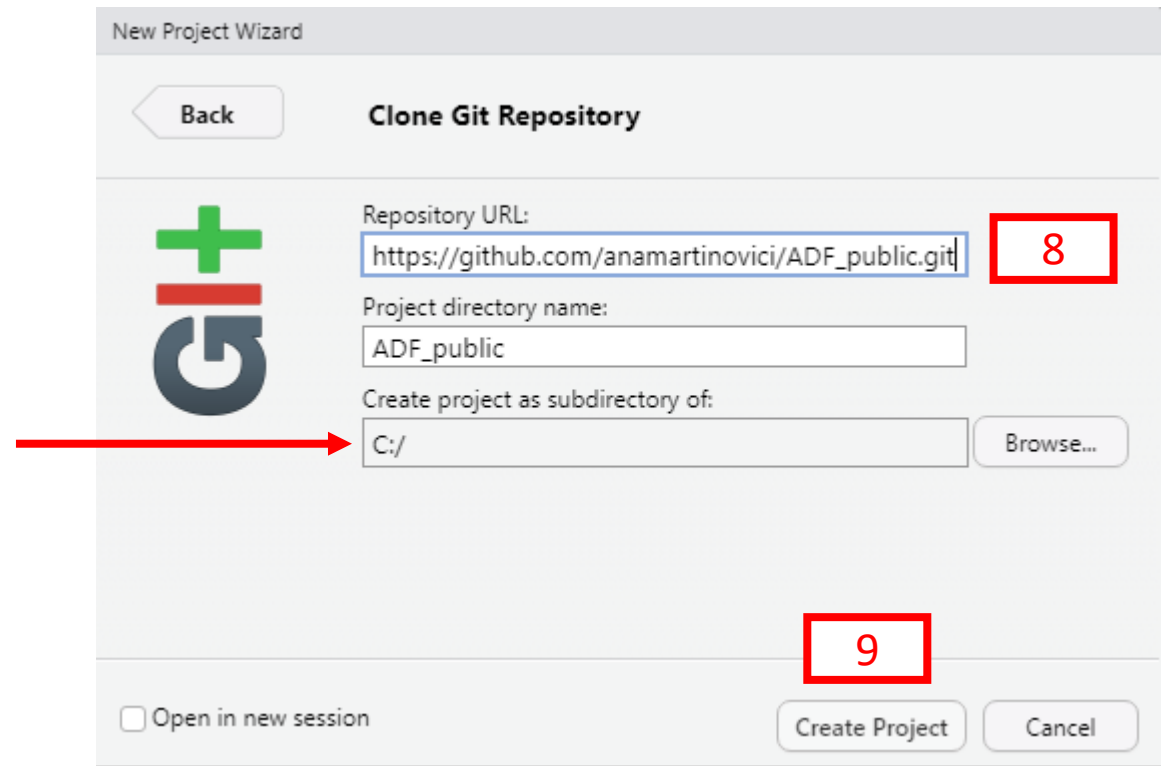


The example uses a repo named ADF\_public, but the steps are the same for other repositories you have access to

8. Paste the [https://github....](https://github.com/anamartinovici/ADF_public.git)  
You've previously copied
9. Create project

Do **NOT** save the repository in Dropbox or other cloud storage options. This will create sync problems that are better avoided. Dropbox and GitHub are useful, but for different purposes.

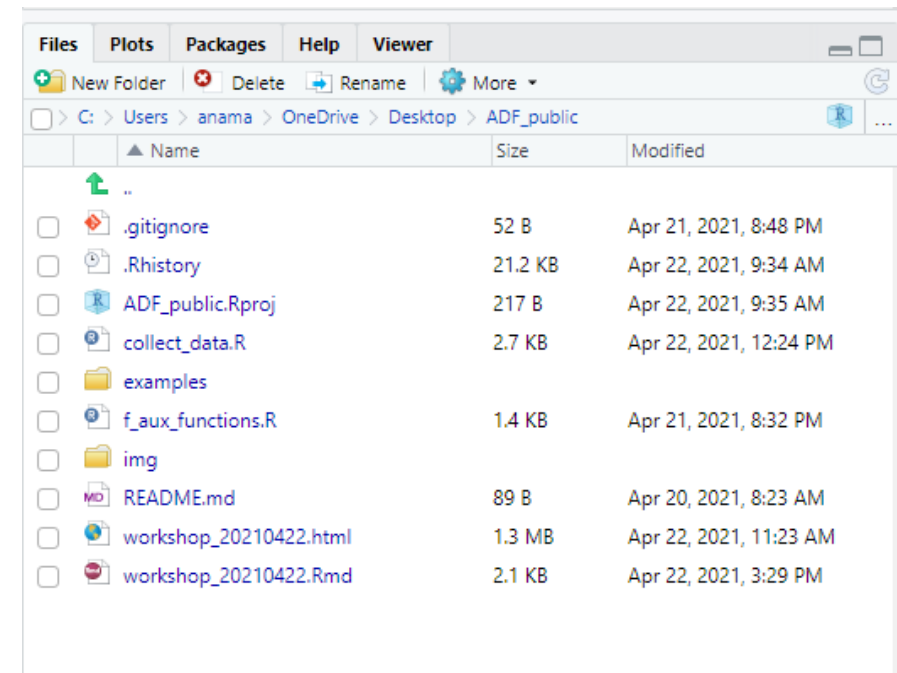
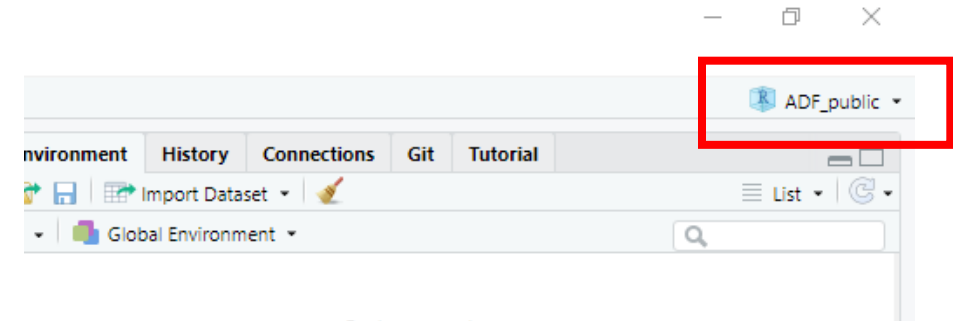
This will create a local folder named "ADF\_public", as a subdirectory of "C:/".



The example uses a repo named ADF\_public, but the steps are the same for other repositories you have access to

After cloning the repo:

- the name of the repo shows in the upper right corner of the RStudio window
- The working directory is set to C:/ADF\_public. Check this by typing “getwd()” in the RStudio console.
- The Files pane will show all the files within the local copy of the repository



# How to clone a repo

**End** of section

# How to add a file to a repo

**Start** of section

# What you need to do to add a file to a remote repository

1. Do you already have a local (see Vocabulary)?
  1. “Yes” -> open the project in RStudio
  2. “No” -> clone the repository on your device
2. Add the file in the folder of to the cloned repository  
This could mean:
  - copy-paste a file you already have in a different folder on your device
  - Create a new file in Rstudio -> save it
3. Stage -> Commit -> Push the file to the remote repository

# How to add a file to a repo

**End** of section

# Recommended way of working with DFCEP\_public\*

**Start** of section

\*or any other public repository you are not a collaborator of



# What should I do with DFCEP\_public?

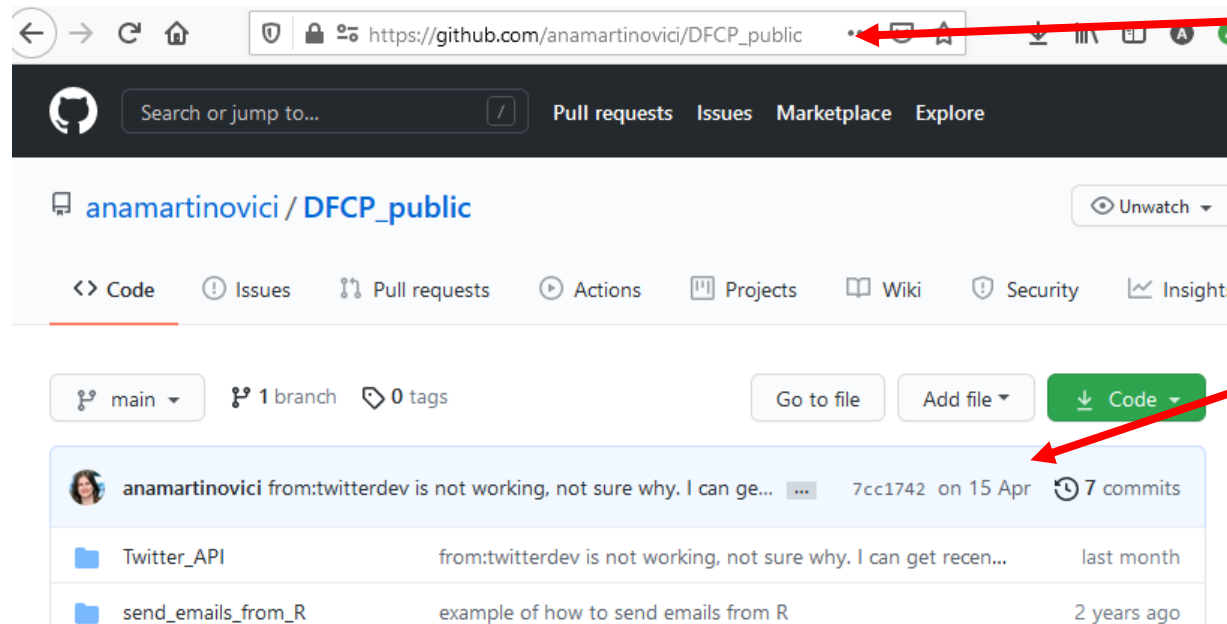
1. Clone it on your laptop. This will make it easier for you to:
  - Go through the example code in Rstudio on your own device
  - Copy paste example code from this repo to your own repository
2. Make sure that if you delete the local repo you are not losing any important work
  - It's ok to modify files within this repo to try things out, but if any of those changes are important for your assignment, then add the changes to your own assignment repo.
  - The next 2 slides explain why this is important.

# The remote can change

- I will make changes to the repository (e.g., add more examples, modify code).

The local you have on your laptop is the version of the repository at the time when you've cloned it. For example: if you clone the repo on May 20 at 18.00, and I push a new file on May 20 at 18.10, then your local copy will not show this file.

- How to check if the remote changed:



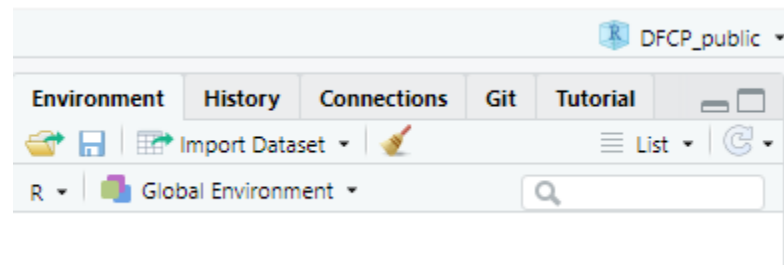
1. Go to the repository

2. Hover the mouse over 15 Apr.  
This will show the date and time of the latest commit.

Click "7 commits" for more info about commits to the repo.  
Click "7cc1742" for more info about the latest commit.

# How to check the time and date for the version you have on your laptop

1. Open the local in RStudio. Check that the name of the repository shows in the upper right corner:



2. Open a terminal and type “git log -1”. The output will be similar to:

```
Console Terminal x Jobs x
Terminal 1 ~ /OneDrive/Desktop/DFCP_public

anama@DESKTOP-MCVPQOA ~ /OneDrive/Desktop/DFCP_public
$ git log -1
commit 7cc1742e048628f0b0ef603616baafe0aaf3a23b (HEAD -> main, origin/main, origin/HEAD)
Author: martinovici@rsm.nl <martinovici@rsm.nl>
Date: Thu Apr 15 15:11:14 2021 +0200

    from:twitterdev is not working, not sure why. I can get recent tweets from other accounts though.
```

# Do the local and remote have the same timestamp?

- “Yes” -> this means I didn’t push any changes to the remote since you’ve cloned the repo.
- “No” -> your local copy is an older version compared to the remote. If you want to get the latest version:
  1. Close RStudio
  2. Delete the local repo
  3. Clone the remote

# Recommended way of working with DFCEP\_public

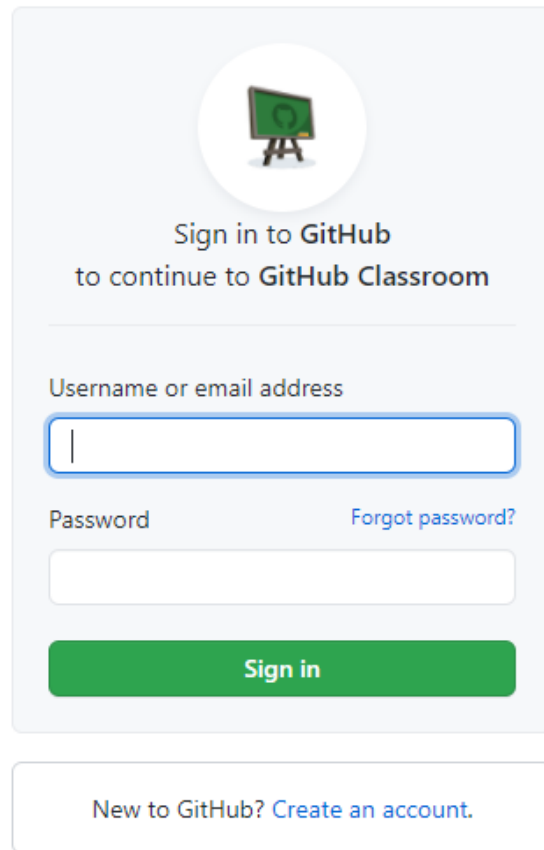
**End** of section

\*or any other public repository you are not a collaborator of

# How to join an assignment (GitHub classroom)

**Start** of section

# Open the invite link in a browser



The image shows a sign-in interface for GitHub Classroom. At the top, there is a circular icon containing a green chalkboard with a white 'G' on it. Below the icon, the text reads "Sign in to GitHub" and "to continue to GitHub Classroom". There is a horizontal line separating this header from the input fields. Below the line, there is a label "Username or email address" followed by a text input field with a blue border. Below that is a label "Password" followed by a text input field. To the right of the password field is a link "Forgot password?". Below the input fields is a green button with the text "Sign in". At the bottom of the form, there is a link "New to GitHub? Create an account."

Sign in to GitHub  
to continue to GitHub Classroom

Username or email address

Password [Forgot password?](#)

**Sign in**

New to GitHub? [Create an account.](#)

This is what you see if you are not already logged in your GitHub account.

Sign in with your GitHub account.

# Open the invite link in a browser

DFCP-classroom-1

Accept the group assignment —

DFCP\_test\_group\_project

Before you can accept this assignment, you must create or join a team. Be sure to select the correct team as you won't be able to change this later.

Create a new team

+ Create team

This is what you see if you are logged in your GitHub account.

Create or join a team —  
**talk to your colleagues beforehand so you know if a team already exists!**



# This is what you see what you create or join a team

You have successfully created team: Cats are the best



DFCP-classroom-1

Accept the group assignment —  
**DFCP\_test\_group\_project**


Accepting this assignment will give your team, **Cats are the best**, access to the `dfcp_test_group_project-cats-are-the-best` repository in the [Digital-Footprints](#) organization on GitHub.

Accept this assignment

# This is what you see after you click “Accept the assignment”



You accepted the assignment, **DFCP\_test\_group\_project** . We're configuring your repository now. This may take a few minutes to complete. Refresh this page to see updates.

 Your assignment is due by **Jun 25, 2021, 18:00 CEST**

Note: You may receive an email invitation to join [Digital-Footprints](#) on your behalf. No further action is necessary.



## Join the GitHub Student Developer Pack

Verified students receive free GitHub Pro plus thousands of dollars worth of the best real-world tools and training from GitHub Education partners — for free. [Learn more](#)

[Apply](#)

# This is what you see after you refresh the page




You're ready to go —

Cats are the best


You accepted the assignment, **DFCP\_test\_group\_project**.

Your team's assignment repository has been created:

 [https://github.com/Digital-Footprints/dfcp\\_test\\_group\\_project-cats-are-the-best](https://github.com/Digital-Footprints/dfcp_test_group_project-cats-are-the-best)

← Click this link to go to the repository

We've configured the repository associated with this assignment ([update](#)).

 Your assignment is due by **Jun 25, 2021, 18:00 CEST**

Note: You may receive an email invitation to join [Digital-Footprints](#) on your behalf. No further action is necessary.

# How to join an assignment (GitHub classroom)

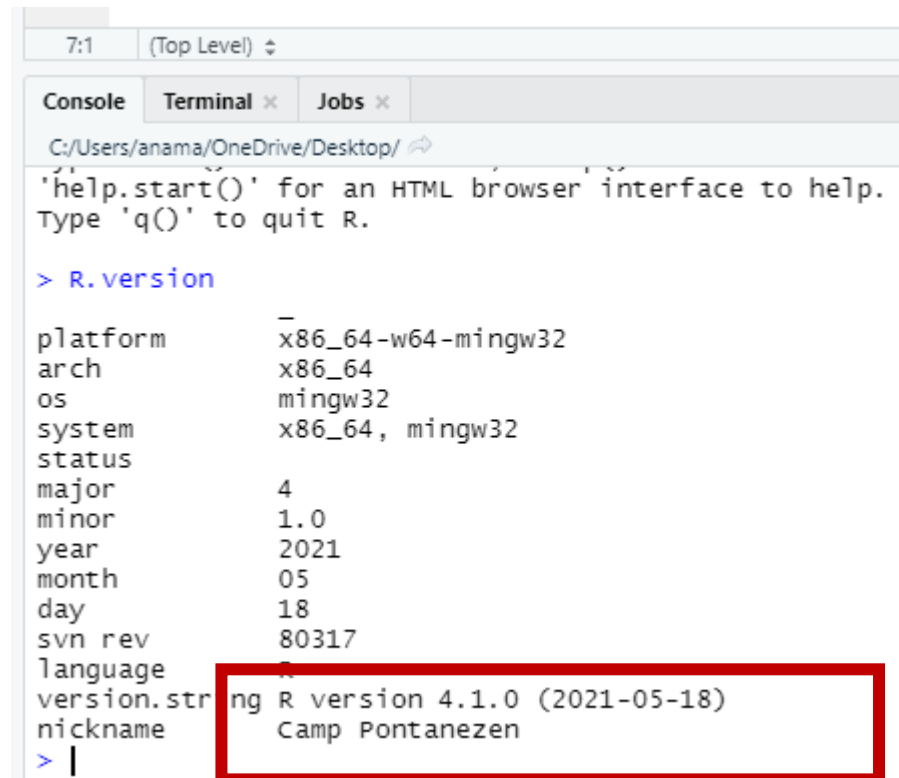
**End** of section

# How to check version number for R and RStudio

**Start** of section

# How to check R version

Type 'R.version' in the console to see which version of R you have on your device. This should be 'R version 4.1.0 (2021-05-18)' called 'Camp Pontanezen'



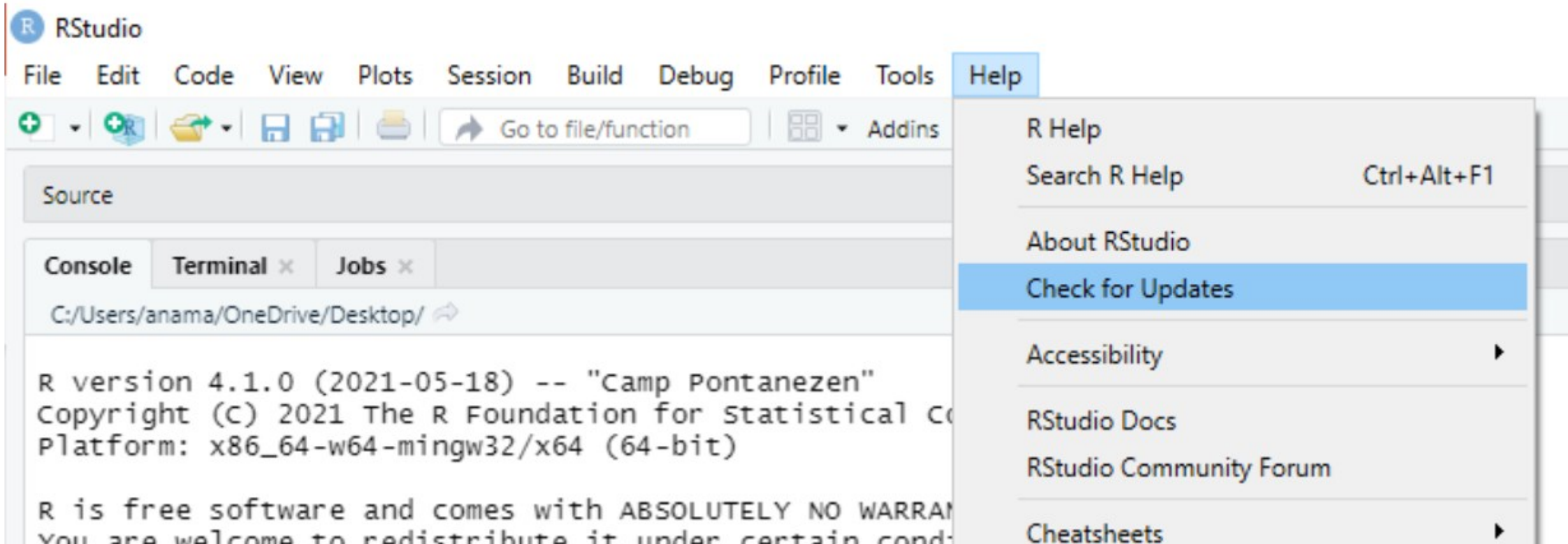
```
7:1 (Top Level) >
Console Terminal x Jobs x
C:/Users/anama/OneDrive/Desktop/
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> R.version

platform      x86_64-w64-mingw32
arch           x86_64
os             mingw32
system         x86_64, mingw32
status
major          4
minor          1.0
year           2021
month          05
day            18
svn rev        80317
language       R
version.string R version 4.1.0 (2021-05-18)
nickname       Camp Pontanezen
> |
```

The screenshot shows the R console interface. At the top, there's a header bar with '7:1 (Top Level)' and a dropdown arrow. Below it are tabs for 'Console', 'Terminal x', and 'Jobs x'. The 'Console' tab is active, showing the current directory 'C:/Users/anama/OneDrive/Desktop/'. The console output displays the result of the 'R.version' command, listing various system and version details. The last two lines, 'version.string R version 4.1.0 (2021-05-18)' and 'nickname Camp Pontanezen', are highlighted with a red rectangular box.

# How to check if you are using the latest RStudio version



# How to check version number for R and RStudio

**End** of section



# How to check if you have git on your device - **Windows**

**Start** of section

# Check if you already have git installed

1. Open RStudio
2. Open a terminal (Tools -> Terminal -> New Terminal)
3. Type “which git”
4. Type “git --version”
5. Check if the output is similar to:

```
anama@DESKTOP-MCVPQOA ~/OneDri  
$ which git  
/c/Program Files/Git/cmd/git
```

```
anama@DESKTOP-MCVPQOA ~/OneDri  
$ git --version  
git version 2.30.0.windows.2
```

```
anama@DESKTOP-MCVPQOA ~/OneDri  
$ █
```

6. If similar, then you have git installed. If you get some error, then you probably don't have git and you need to install it.

# If you don't already have git on your device, then install it

- Follow the instructions here: <https://happygitwithr.com/install-git.html#install-git-windows>
  - There are multiple options for Windows, choose one of them.
- After you've followed the steps to install git, restart your device, and check again if git is installed (using “which git” and “git --version” commands on the previous slide).

# How to check if you have git on your device - **Windows**

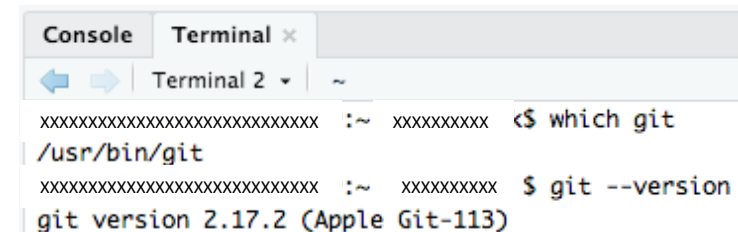
**End** of section

# How to check if you have git on your device - MacOS

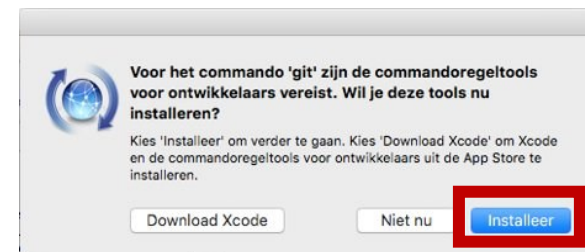
**Start** of section

# Check if you already have git installed

1. Open RStudio
2. Open a terminal (Tools -> Terminal -> New Terminal)
3. Type “which git”
4. Type “git --version”
5. Check if the output is similar to:
  - If the output is similar, then you have git installed on your device.
6. If you get a notification saying you don't have git and asking you to install it, then click 'Install'.



```
Console Terminal x
Terminal 2 ~
XXXXXXXXXXXXXXXXXXXXXXXXXXXXX :~ XXXXXXXXXX <$ which git
/usr/bin/git
XXXXXXXXXXXXXXXXXXXXXXXXXXXXX :~ XXXXXXXXXX $ git --version
git version 2.17.2 (Apple Git-113)
```



# How to check if you have git on your device - MacOS

**End** of section

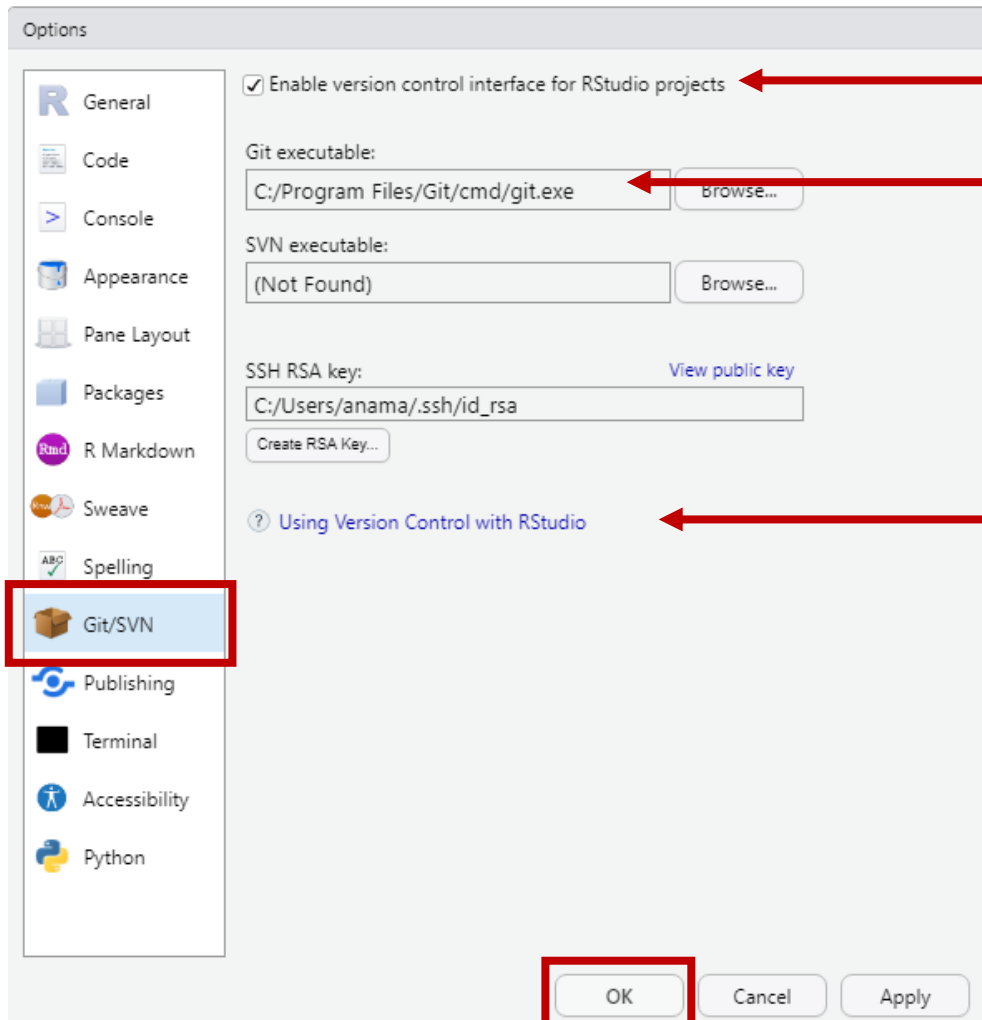
# How to check the path to git.exe in RStudio

**Start** of section



# Check the path to git.exe

RStudio -> Tools -> Global Options. Then click on 'Git/SVN'.



Make sure that this is selected

This should be the same path as the output of "which git"

If you want more info, click here

Click 'Ok'. A notification may appear to restart RStudio. If this is the case, please do so.

# How to check the path to git.exe in RStudio

**End** of section

# How to introduce yourself to GitHub

**Start** of section

# First use of git

- If you have git on your device, the next step is to tell your device how to introduce yourself to GitHub. You will do this by using your GitHub account, so GitHub will recognize you as a user and more importantly as a contributor to repositories you want to commit and push to.
- RStudio -> Tools -> Shell -> type the following commands:

```
git config --global user.name 'Jane Doe'
```

```
git config --global user.email 'jane@example.com'
```

Replace 'Jane Doe' and 'jane@example.com' with the username and email address that you used for your GitHub account

```
git config --global --list
```

This command prints the user.name and user.email information. Check if it is correct.

# How to introduce yourself to GitHub

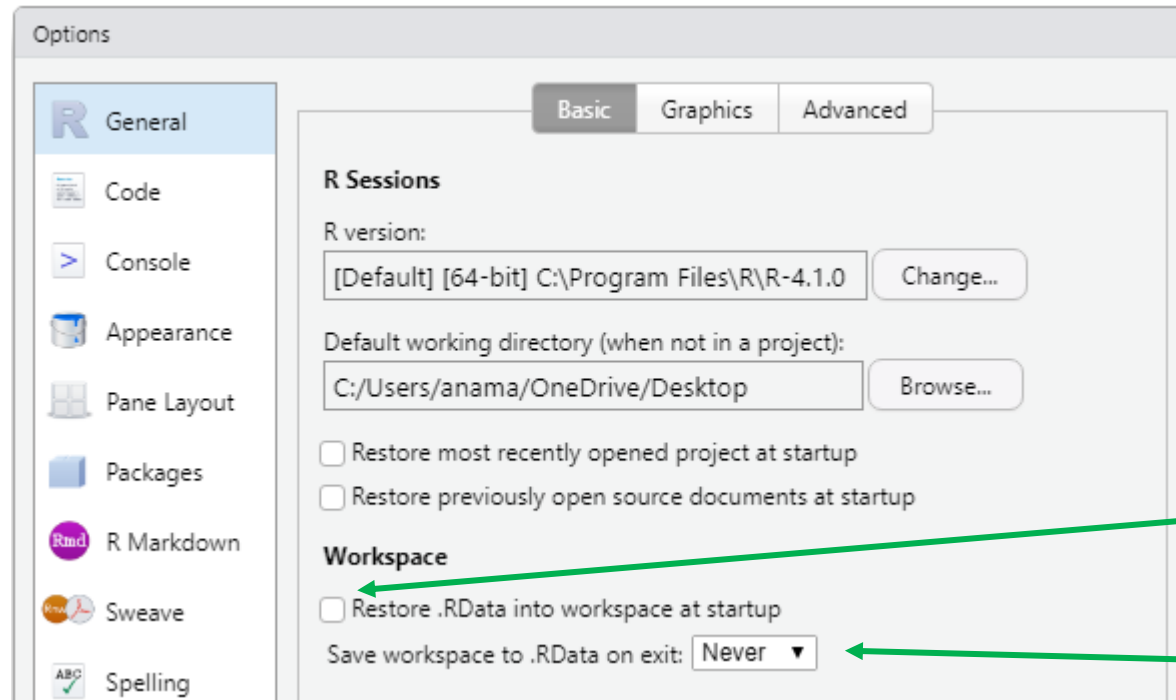
**End** of section

# RStudio settings

**Start** of section

# Make sure the options you have match what is on the slide (for restoring data at startup and saving data on exit)

Rstudio -> Tools -> Global Options



Do NOT restore .RData at startup

Do NOT save .RData on exit

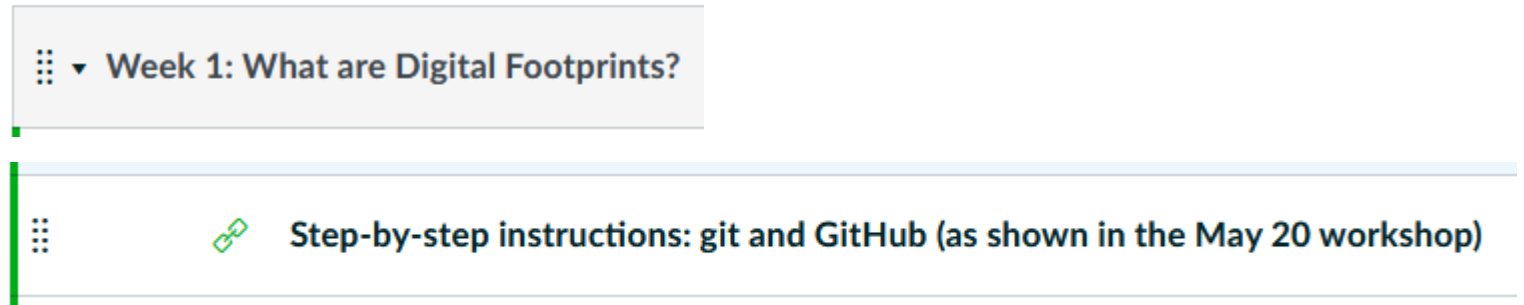
# RStudio settings

**End** of section



# Set 2

You find the Canvas version under Modules ->

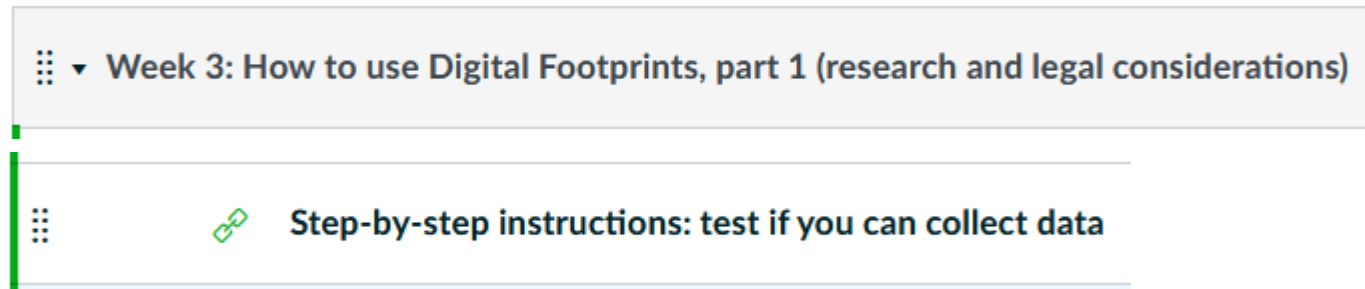


**End**



# Set 3

You find the Canvas version under Modules ->



**Start**

# Step-by-step instructions

## Week 3

Dr. Ana Martinovici

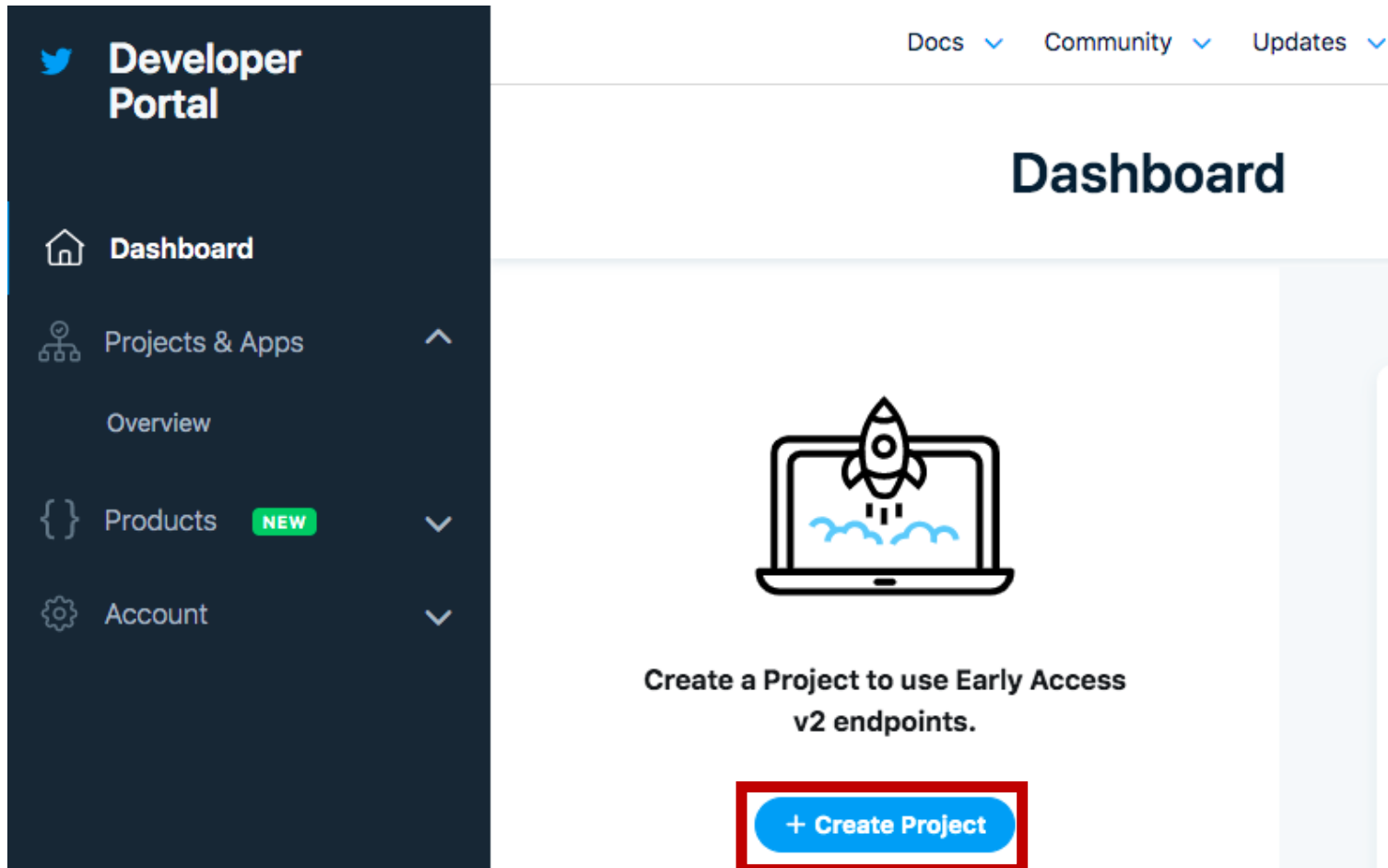
# This file gives step-by-step instructions for:

- Creating an App in the Twitter developer platform
- Saving the bearer token on your device
- Testing your connection to the API

# How to create an app in the Twitter developer platform

**Start** of section

# Create a project and app in the Twitter Dashboard



Click '+ Create Project'  
**or**  
click 'Projects & Apps'  
then 'Overview' followed  
by '+ New Project'

# Create a project and app in the Twitter Dashboard

- Give your Project a name and click 'Next'
  - You can pick whatever name you would like



## Name your Project

Your Project helps you organize your work and monitor your usage with the Twitter API.


80

Next



# Create a project and app in the Twitter Dashboard

- Which best describes you? Pick 'Student' and click 'Next'



**Which best describes you?**

Below you will find options on how you intend to use the Twitter Developer Platform.

Student

**Next**

Back

STEP 1

STEP 2

# Create a project and app in the Twitter Dashboard

- Describe why you are making a new project and click 'Next'
- You may use the example answer below



## Describe your new Project

This info is just for us, here at Twitter. It'll help us create better developer experiences down the road.

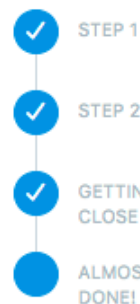
This project is for a university course assignment.



Back

# Create a project and app in the Twitter Dashboard

- The last step is giving your App a name and then click on 'Complete'
  - You can pick whatever name you would like



## Last step, name your App

Apps are where you get your **access keys and tokens** and set permissions. They are encompassed within your Projects.

32

Complete

Back

# Create a project and app in the Twitter Dashboard

- After you clicked 'Complete', your keys and tokens will appear. For safety reasons, Twitter shows your keys and tokens only once. **Copy paste them into a safe location.** If you need to regenerate them, check the instructions at the end of this document.
- You can hover your mouse over the “i” sign to get a brief explanation.
- For more details:  
<https://developer.twitter.com/en/docs/authentication/oauth-2-0/bearer-tokens>



## Here are your keys & tokens

For security, this will be the last time we'll display these. If something happens, you can always regenerate them.

### API key ⓘ

5K XXXXXXXXXXXXXXXX jA2



### API secret key ⓘ

KE XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX jAJ  
J7



### Bearer token ⓘ

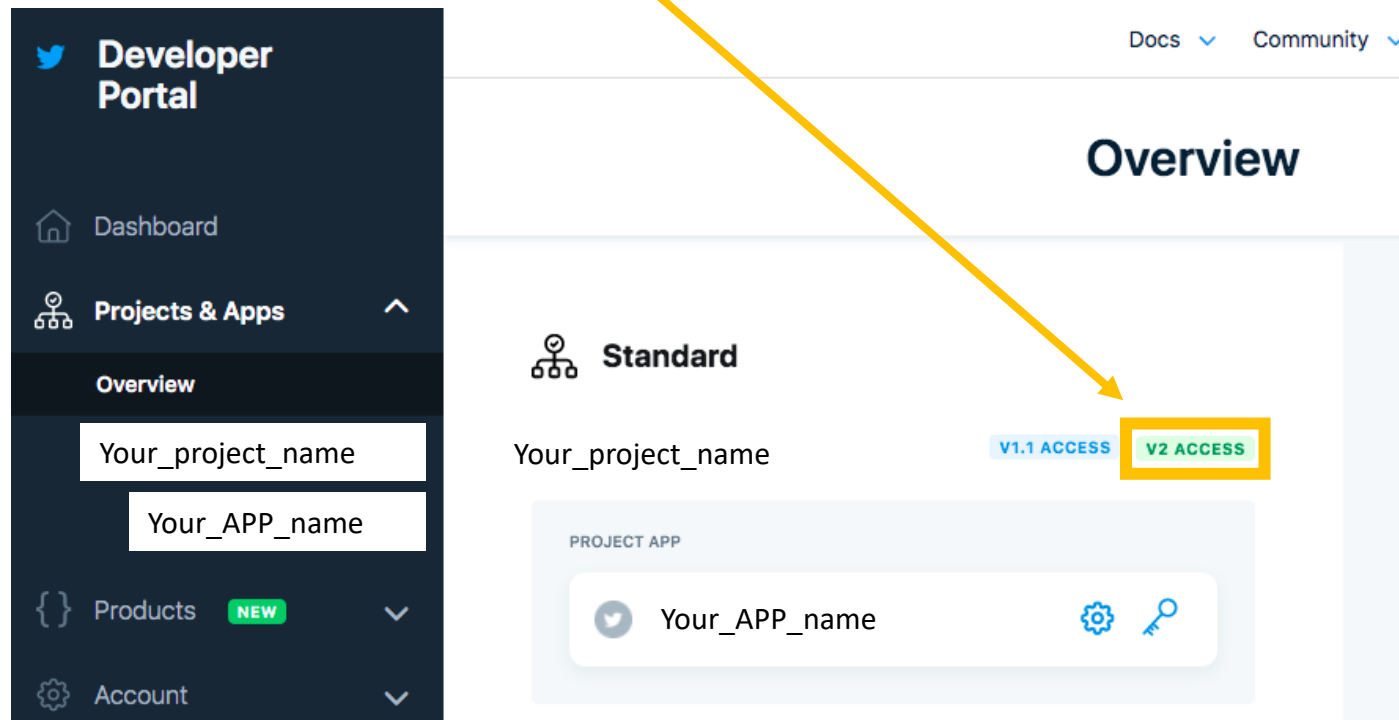
AA XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX %2Fv  
M6 XXXXXXXX 'UITYJ  
hLs



# Create a project and app in the Twitter Dashboard

Now if you check 'Projects & Apps' -> 'Overview', you will see your project!

It is important that you see 'V2 Access', as you will need this to collect Twitter data.



# How to create an app in the Twitter developer platform

**End** of section

# How to save the bearer token on your device

**Start** of section

# What is a bearer token?

- To keep it brief, you need a bearer token to authenticate requests you make from the Twitter API.
- The bearer token works a bit like a unique password that tells the Twitter API that you are allowed to collect data. If the API recognizes your token and your token has the right to collect the data you've requested, then the API will give you the data. Otherwise, the API doesn't give you data.
- For more details: <https://developer.twitter.com/en/docs/authentication/oauth-2-0>



# How to use the bearer token

- For your assignment, you will add code files to the assignment repository on GitHub. It is not smart to push tokens or passwords to repositories, even if they are private repositories. You shouldn't share the token with anyone else, not even the other colleagues in your team!
- The smart thing to do is to create an environment variable that stores your bearer token. This way, all the R sessions on your device will know the bearer token. This way, the bearer token will not show up in any of the files you're pushing to GitHub, so it stays secret – as it should be.

# If you are curious about environment variables in R

Environment variables affect all R sessions on a specific device. For more info:

- <https://stat.ethz.ch/R-manual/R-devel/library/base/html/EnvVar.html>
- Chapter 7 of “What they forgot to teach you about R” (<https://rstats.wtf/r-startup.html>)

You don't need to understand all the details around environment variables. What's important for now is that you can use an environment variable to store your Twitter API bearer token.

# From “What they forgot to teach you about R” (<https://rstats.wtf/r-startup.html>, Chapter 7)

## 7.1 .Renvi ron

The `.Renvi ron` file is most useful for defining sensitive information such as API keys (such as GitHub or twitter) as well as R specific environment variables like the history size ( `R_HISTSIZE=100000` ) and default library locations `R_LIBS_USER` .

The `.Renvi ron` file contains lists of environment variables to set. This is *not* R code, it uses a format similar to that used on the command line shell.

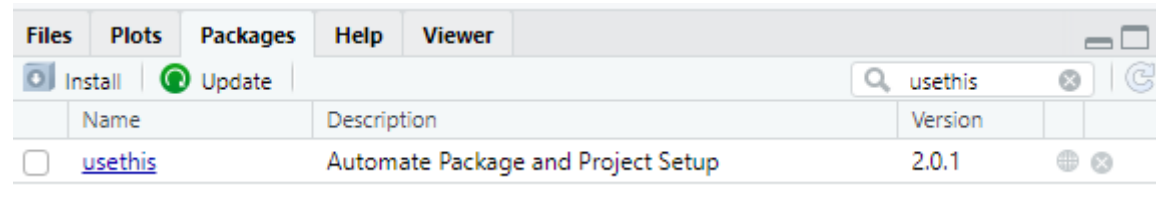
The easiest way to edit `.Renvi ron` is by running `usethis::edit_r_envi ron()` .

A simple example of a `.Renvi ron` file is

```
R_HISTSIZE=100000
GITHUB_PAT=abc123
R_LIBS_USER=~ /R/%p/%v
```

# The “easiest” way to edit environment variables

1. Make sure you have the “usethis” R package. If you don’t yet have the package, then install it. I have version 2.0.1:



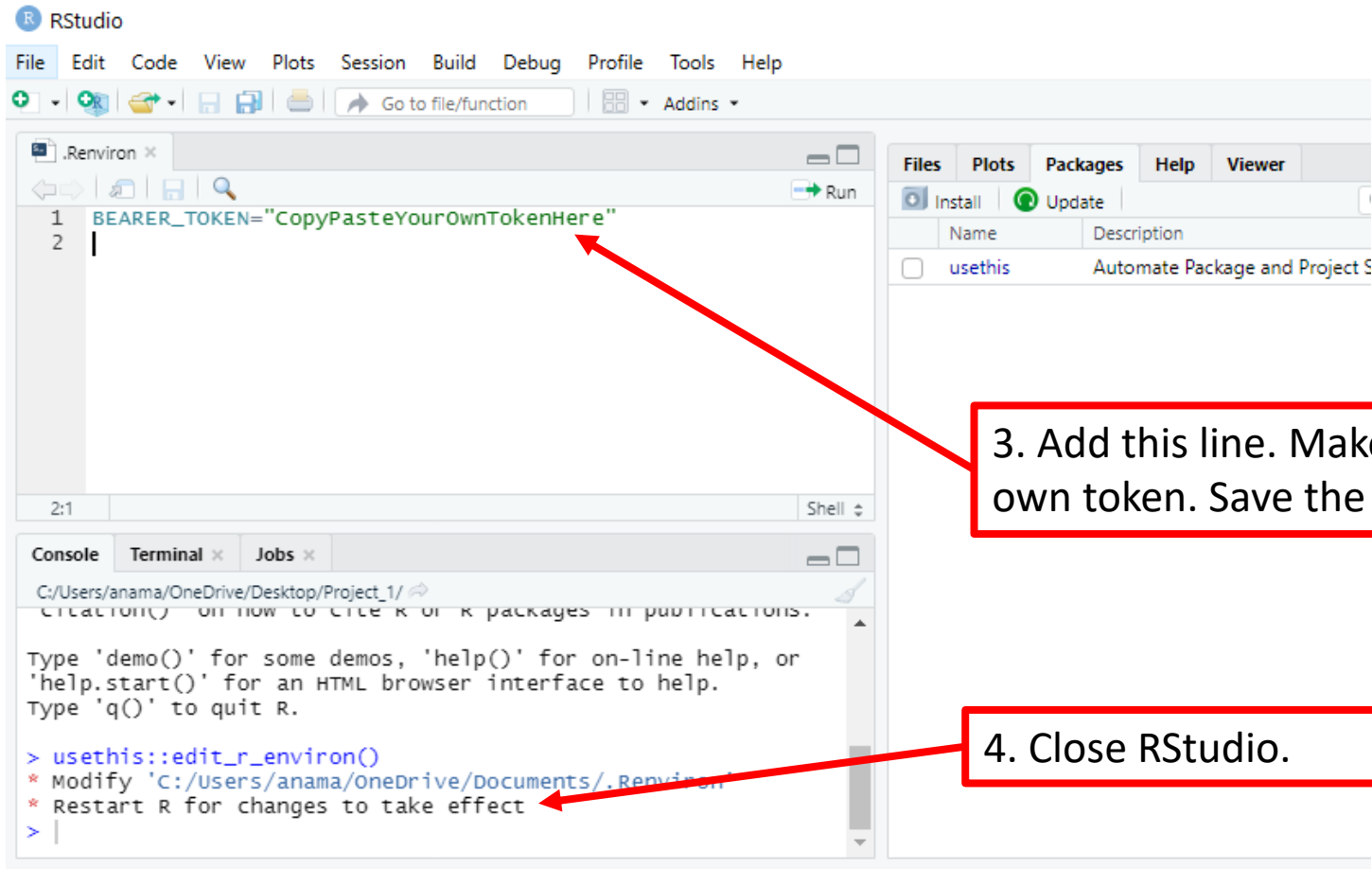
2. Type “usethis::edit\_r\_environ()” in the console:

```
> usethis::edit_r_environ()  
* Modify 'C:/Users/anama/OneDrive/Documents/.Renviron'  
* Restart R for changes to take effect  
|
```

This is the location of the .Renviron file. If you don’t yet have this file, it will create it for you.

# The “easiest” way to edit environment variables

This is what you should see in RStudio:



# The “easiest” way to edit environment variables

5. Open RStudio
6. Type “Sys.getenv(“BEARER\_TOKEN”)” in the console:

```
type q() to quit R.  
  
> Sys.getenv("BEARER_TOKEN")  
[1] "CopyPasteYourOwnTokenHere"  
> |
```

---

You should see the bearer token you’ve added to the .Renviron file. The screenshot shows “CopyPasteYourOwnTokenHere” because that’s what I’ve added (see previous slide).

# How to save the bearer token on your device

**End** of section

# How to test your connection to the API

**Start** of section



# Have you already joined the individual practice repository on GitHub?

If “yes”, then move to the next slide of this presentation.

If “no”, then:

- Check slide 9 of the  [Step-by-step instructions: git and GitHub \(as shown in the May 20 workshop\)](#)

You find this file on Canvas under Modules -> Week 1.

If “not sure”, then:

- Login to GitHub Classroom and see if you can find the repo. For details on what name this repo should have, check the same slide 9.

# Clone these two repositories:

- DFCP-practice-YourGitHubUserName
- DFCP\_public

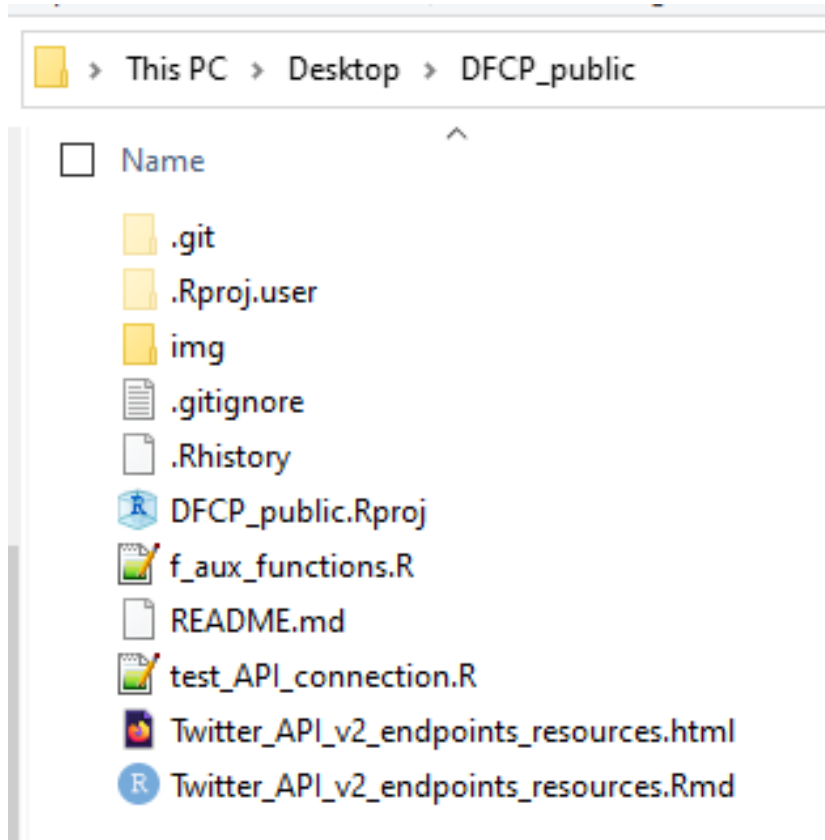
For step-by-step instructions on how to clone a repo, check



Step-by-step instructions: git and GitHub (as shown in the May 20 workshop)

You find this file on Canvas under Modules -> Week 1.

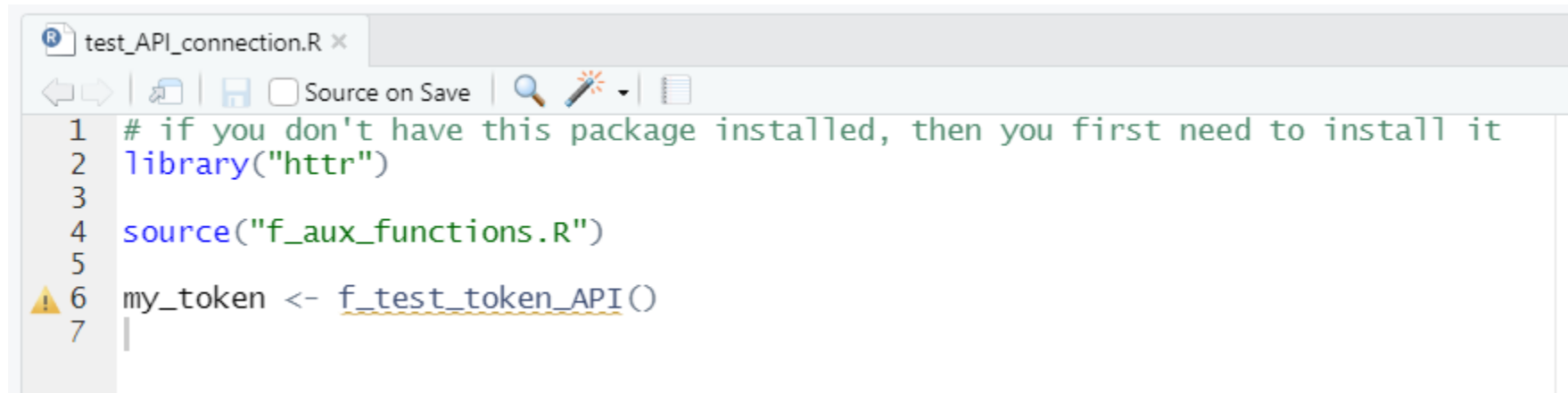
# Open the folder that contains the local copy of DFCP\_public



- On my laptop this is on Desktop, but on your device this can be somewhere else
- Copy the “f\_aux\_functions.R” and “test\_API\_connection.R” files and paste them in the folder that contains the local copy of DFCP-practice-YourGitHubUserName

# Open DFCP-practice-YourGitHubUserName in RStudio

- Open “test\_API\_connection.R”
- Select all code (CTRL+A / Cmd+A) and then execute the code (CTRL+Enter / Cmd+Return)



```
test_API_connection.R x
Source on Save
1 # if you don't have this package installed, then you first need to install it
2 library("httr")
3
4 source("f_aux_functions.R")
5
6 my_token <- f_test_token_API()
7
```

# Check the output in the console

- If it is similar to this, then you can successfully get data from the API

```
> # if you don't have this package installed, then you first need to install it
> library("httr")
>
> source("f_aux_functions.R")
>
> my_token <- f_test_token_API()
The bearer token has a value. Let's see if it's the correct value.
The HTTP status code is: 200
This means Success!
> |
```

- Otherwise, check what the message says and try to fix the issue. If you don't know how to fix it, let me know and I'll help you.

# Final step:

- Commit and push the files you've just added to your repository.
- This way, it's easy to let me know if you can connect to the API or not. Use the Issues to send me a message. If you've had issues connecting to the API, creating a new Issue also makes it easier for me to help you fix errors in your code.

See <https://docs.github.com/en/issues/tracking-your-work-with-issues/creating-issues/creating-an-issue> for info about how to create an issue.

# How to test your connection to the API

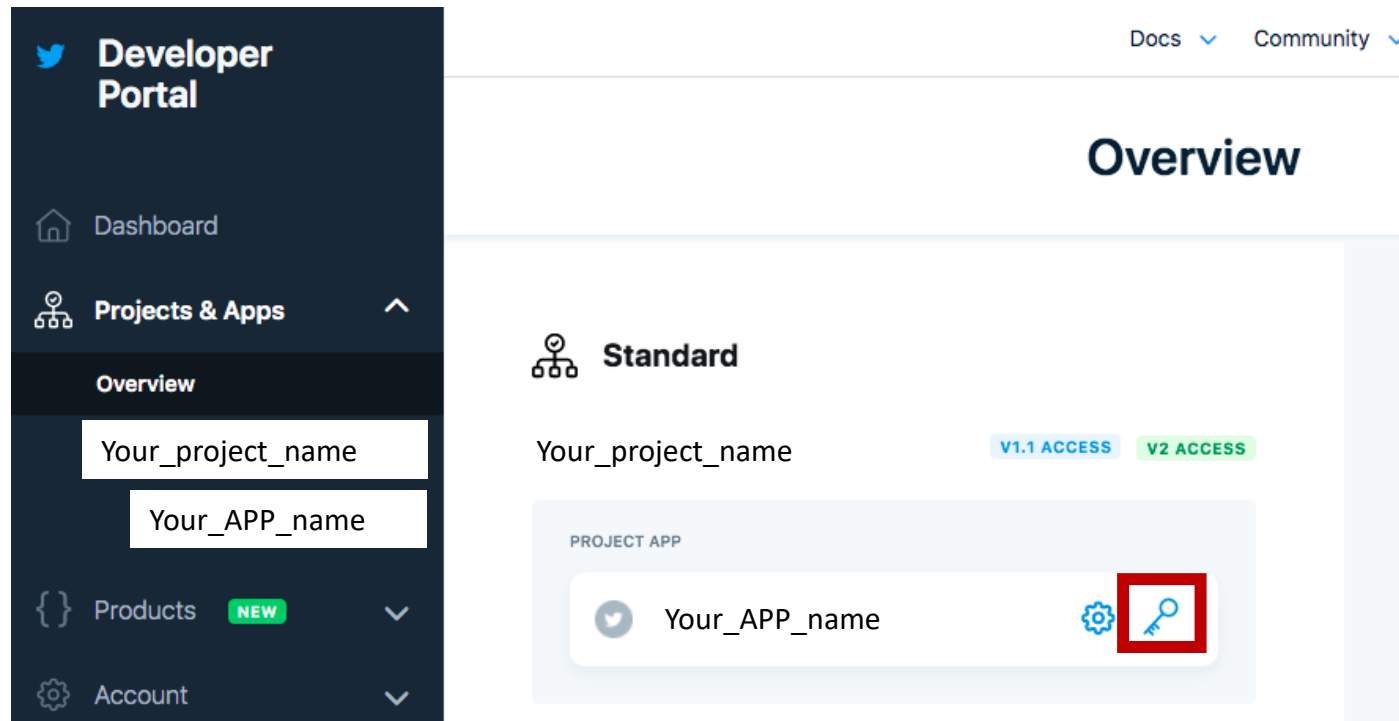
**End** of section

# How to regenerate the bearer token

**Start** of section



Twitter shows the keys and tokens only one time and it is important to store them somewhere where you can easily find them.  
If you do lose your keys and tokens, click the key sign.



Here you can click on 'Regenerate' for the keys or tokens that you want to replace.

Developer Portal

Dashboard

Projects & Apps

Overview

Your\_project\_name

Your\_APP\_name

Products

Account

NEW

Docs

Community

Your\_project\_name

Your\_APP\_name

Settings

Keys and tokens

Consumer Keys

API key & secret

Regenerate

Authentication Tokens

Bearer token

Generated March 13, 2021

Regenerate

Revoke

Access token & secret

For

Your\_account\_name

Generate

Click 'Yes, regenerate' and make sure to save the newly created keys or token. Once you have done this, you can click on 'Yes, I saved it'.

Are you sure you want to regenerate?

If you regenerate your bearer token, you might need to update your code for your app to work properly.

Want to regenerate?

Cancel

Yes, regenerate

Here is your new bearer token. Have you saved it?

For security, this will be the last time we'll display this. If something happens, you can always regenerate it.

Bearer token:

AA/ XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

NiS XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Cml XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

JARe

BTJC

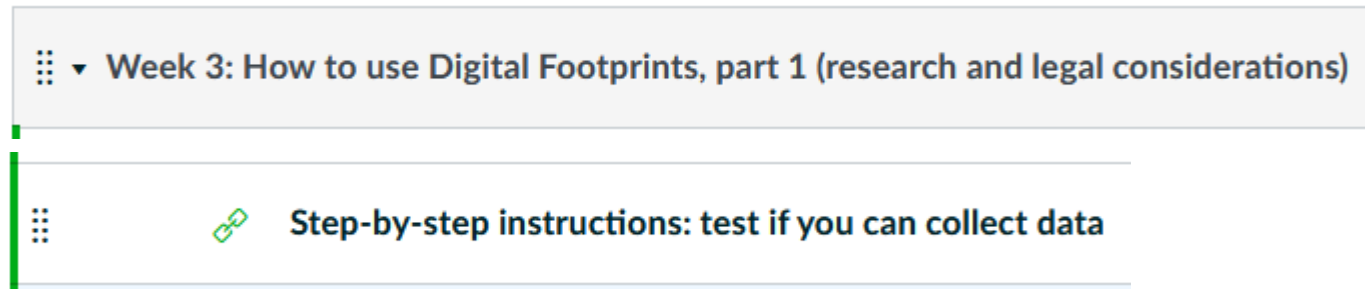
Yes, I saved it

# How to regenerate the bearer token

**End** of section

# Set 3

You find the Canvas version under Modules ->



**End**