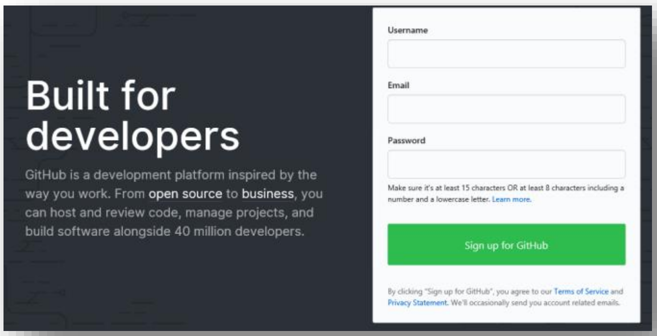
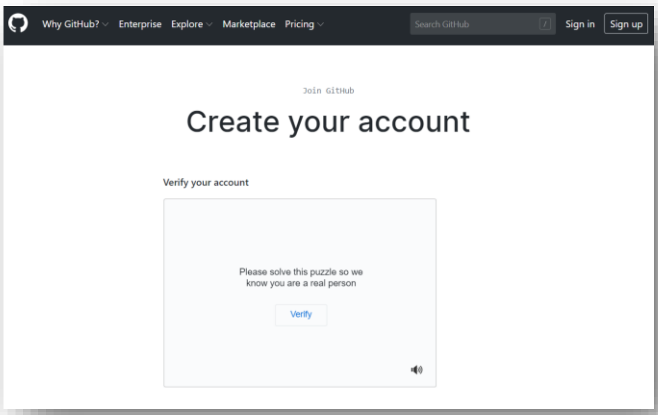
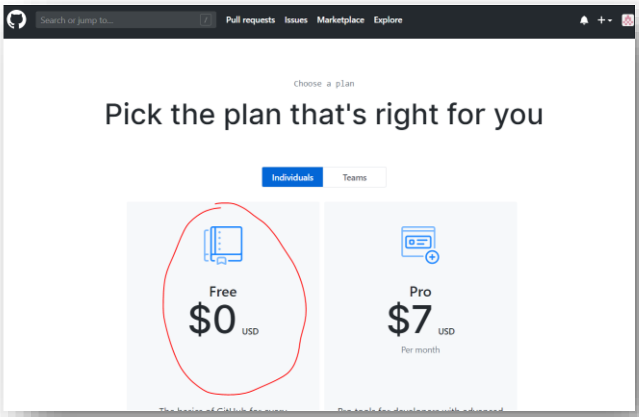
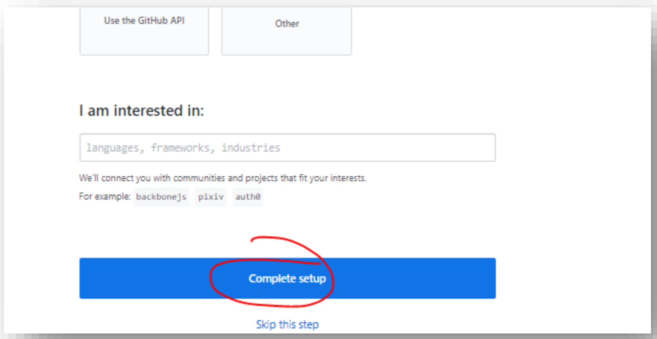
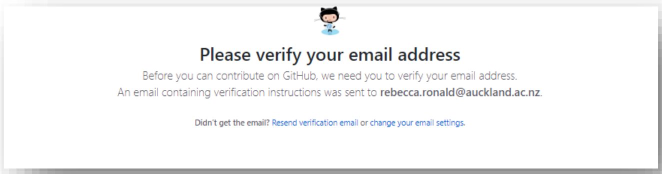
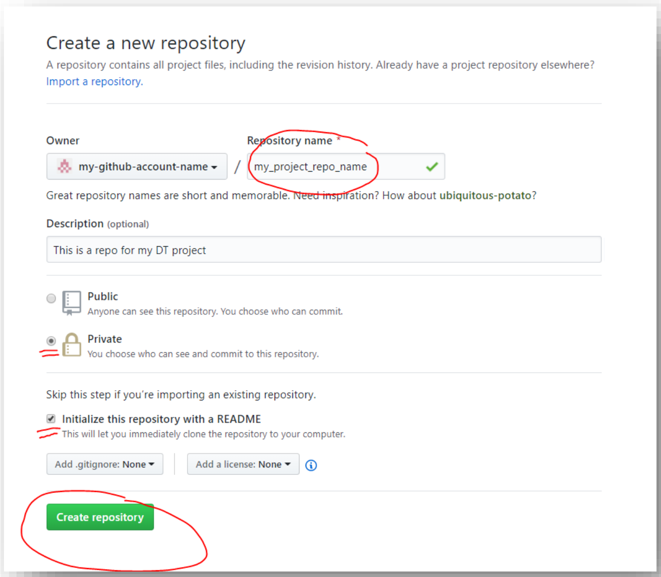
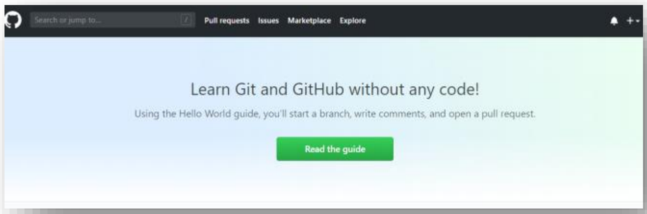
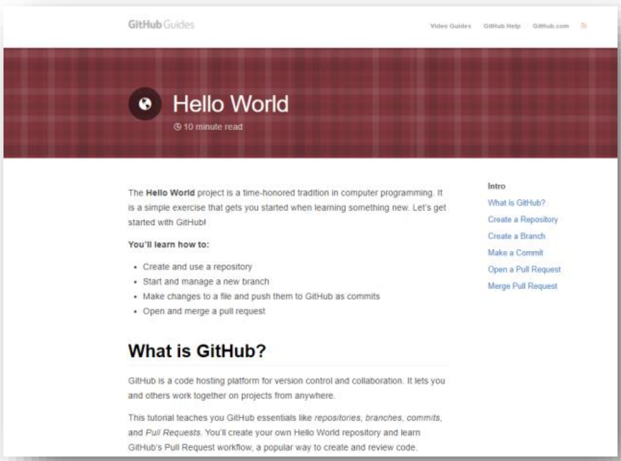
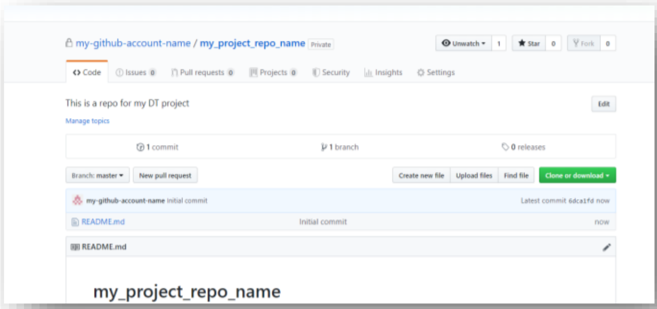
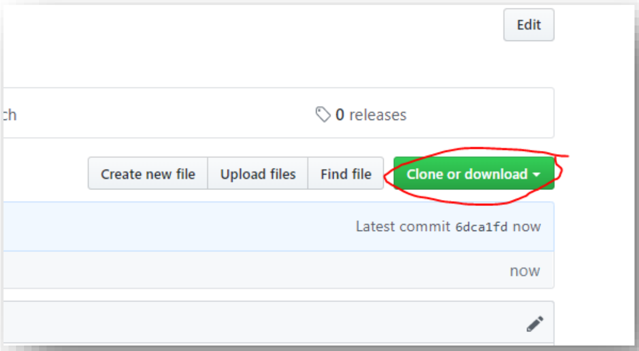
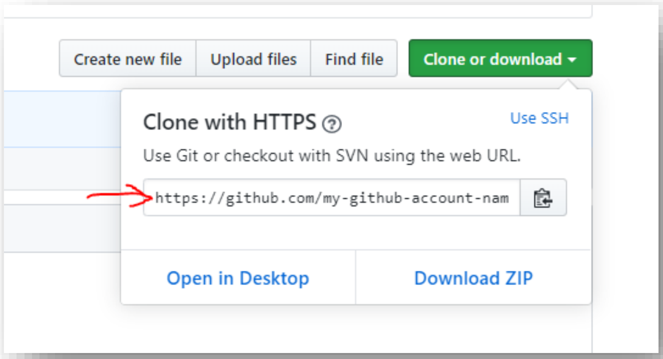
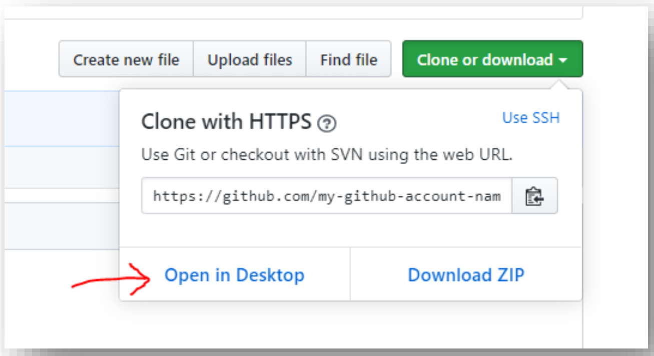
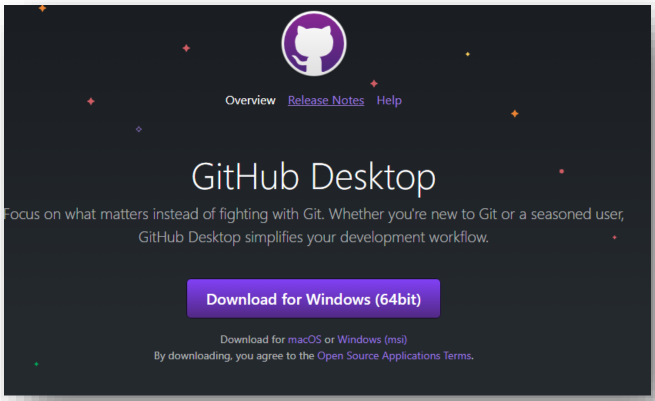
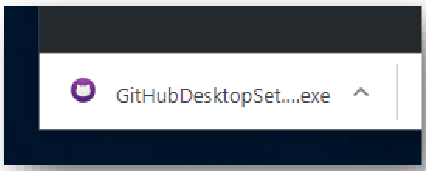
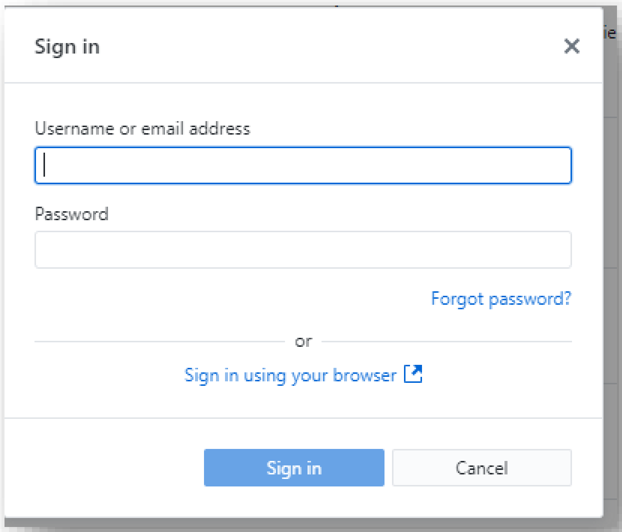
GitHub and GitHub Desktop for Beginners

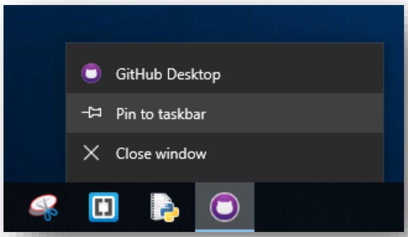
Log in and initial setup	
Create a login to GitHub at https://github.com	 Built for developers GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside 40 million developers. Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. Learn more. By clicking "Sign up for GitHub", you agree to our Terms of Service and Privacy Statement. We'll occasionally send you account related emails.  Why GitHub? Enterprise Explore Marketplace Pricing Search GitHub Sign in Sign up Join GitHub Create your account Verify your account Please solve this puzzle so we know you are a real person
Select the free plan and complete setup.	 Search or jump to... Pull requests Issues Marketplace Explore Choose a plan Pick the plan that's right for you Individuals Teams Free $0 USD Pro $7 USD Per month

	 I am interested in: languages, frameworks, industries We'll connect you with communities and projects that fit your interests. For example: backbonejs pixiv auth0
You will be prompted to verify your email address. Check your emails and click the button to verify.	 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 **rebecca.ronald@auckland.ac.nz**. Didn't get the email? Resend verification email or change your email settings.
You will be prompted to create a new repository. Give it a name, set it to PUBLIC to make it easier to share it with your teacher, initialise it with a README file (this is good practice), and click 'Create repository'.	 Create a new repository A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository. Owner: my-github-account-name / Repository name *: my_project_repo_name Great repository names are short and memorable. Need inspiration? How about **ubiquitous-potato**? Description (optional): This is a repo for my DT project Public — Anyone can see this repository. You choose who can commit. [✓ Private — You choose who can see and commit to this repository.] Skip this step if you're importing an existing repository. [✓ Initialize this repository with a README] — This will let you immediately clone the repository to your computer. Add .gitignore: None Add a license: None
During your setup, you will see a link to learn GitHub. This is optional – it is a good way to get familiar with the interface and with branches, but if you will be using GitHub mostly locally, you don't need this so much.	 Search or jump to... Pull requests Issues Marketplace Explore Learn Git and GitHub without any code! Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

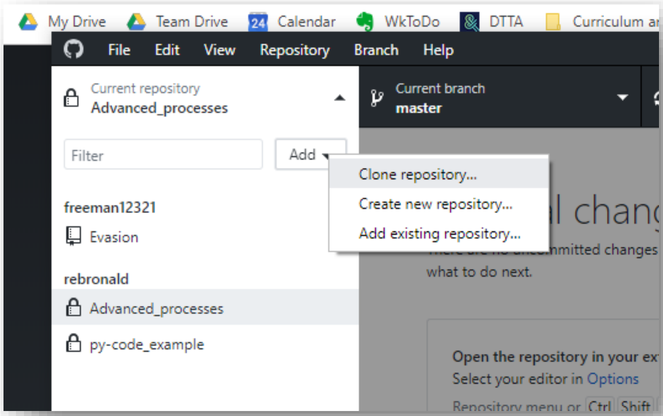
	 GitHub Guides Video Guides GitHub Help GitHub.com **Hello World** 10 minute read The **Hello World** project is a time-honored tradition in computer programming. It is a simple exercise that gets you started when learning something new. Let's get started with GitHub! **You'll learn how to:** • Create and use a repository • Start and manage a new branch • Make changes to a file and push them to GitHub as commits • Open and merge a pull request **What is GitHub?** GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. This tutorial teaches you GitHub essentials like *repositories*, *branches*, *commits*, and *Pull Requests*. You'll create your own Hello World repository and learn GitHub's Pull Request workflow, a popular way to create and review code. Intro What is GitHub? Create a Repository Create a Branch Make a Commit Open a Pull Request Merge Pull Request
Now you have your empty repo (repository) set up, you can make the link to a local program and your documents folders.	 my-github-account-name / my_project_repo_name Private Unwatch 1 Star 0 Fork 0 Code Issues 0 Pull requests 0 Projects 0 Security Insights Settings This is a repo for my DT project Edit Manage topics 1 commit 1 branch 0 releases Branch: master New pull request Create new file Upload files Find file Clone or download my-github-account-name Initial commit Latest commit 6dca1fd now README.md Initial commit now README.md my_project_repo_name
Click on 'Clone or download'.	 Edit 0 releases Create new file Upload files Find file Clone or download Latest commit 6dca1fd now now
The next step depends on what program you are using to manage the GitHub files on your computer. If you are working within another program, such as Atom, Visual Studio Code or Git Bash you will need the URL. If you want to install and use GitHub Desktop you can use 'Open in Desktop'	 Create new file Upload files Find file Clone or download Clone with HTTPS Use SSH Use Git or checkout with SVN using the web URL. `https://github.com/my-github-account-nam` Open in Desktop Download ZIP

	 Create new file Upload files Find file Clone or download Clone with HTTPS Use SSH Open in Desktop Download ZIP Use Git or checkout with SVN using the web URL. `https://github.com/my-github-account-nam`
If using GitHub Desktop	
If you are using the GitHub Desktop option, a few more steps. Clicking on the 'Open in Desktop' link in your repo will take you to a download page. Download the appropriate version for your machine.	 Overview Release Notes Help GitHub Desktop Download for Windows (64bit) Focus on what matters instead of fighting with Git. Whether you're new to Git or a seasoned user, GitHub Desktop simplifies your development workflow. Download for macOS or Windows (msi) By downloading, you agree to the Open Source Applications Terms.
Click to run the install file…	
Sign in using the GitHub credentials that you created for the GitHub website.	 Sign in Username or email address Password Forgot password? or Sign in using your browser Sign in Cancel

Pinning GitHub Desktop to your taskbar isn't a bad idea, if you will be using it a lot.

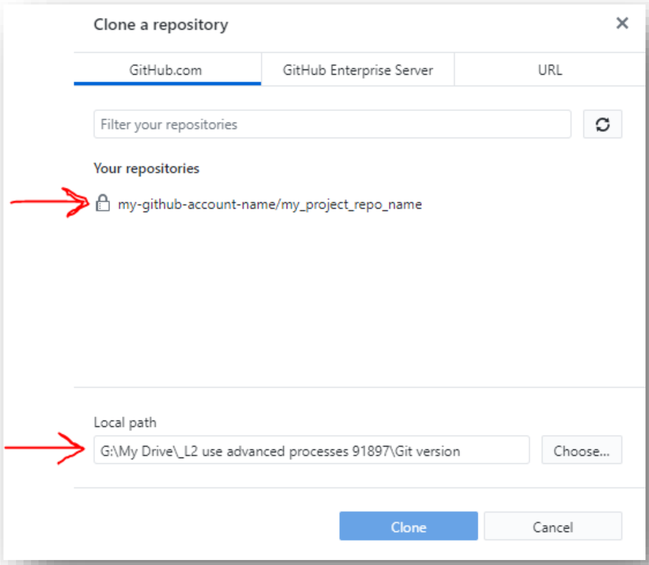


You need to click on the Add button and clone your online repository to this computer.



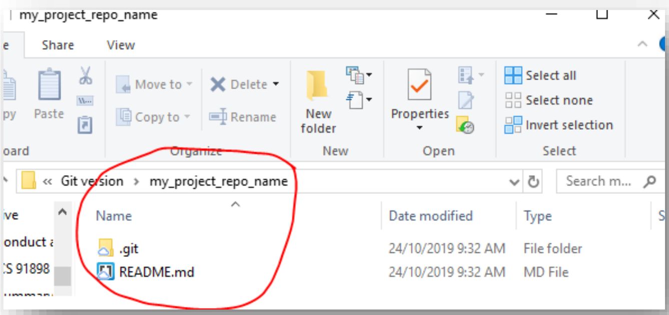
You need to choose the repo you want to clone. You also need to choose where on your computer you want to save it. Choose a place that you can work from easily – eg your Documents or Google Drive File Stream.

Creating a new folder named after your project is a good way to keep things tidy. Clicking the blue 'Clone' button will set up a .git folder in your project folder, and add the README file (and any other files in your online GitHub repo) to it.



my-github-account-name/my_project_repo_name

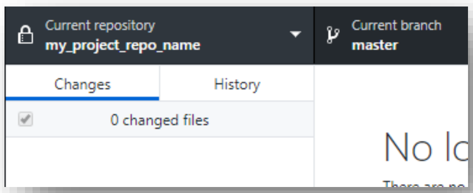
Local path: G:\My Drive_L2 use advanced processes 91897\Git version

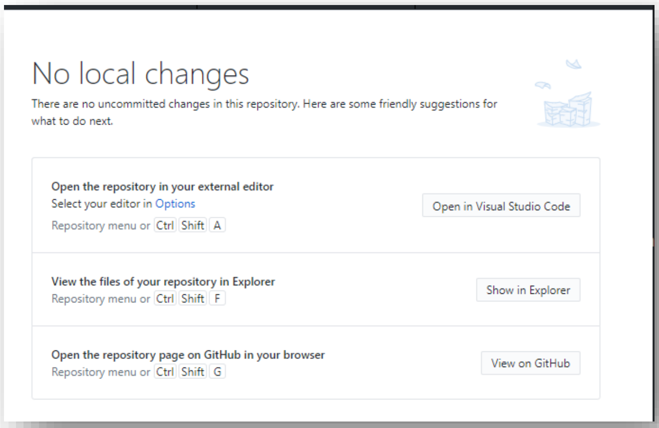


Name	Date modified	Type
.git	24/10/2019 9:32 AM	File folder
README.md	24/10/2019 9:32 AM	MD File

You should now be working in your new repo. Check under ‘Current repository’ at the top left of your GitHub Desktop window.

Your main screen should tell you that there are no changes yet – this means that your local version matches the online one.




No local changes

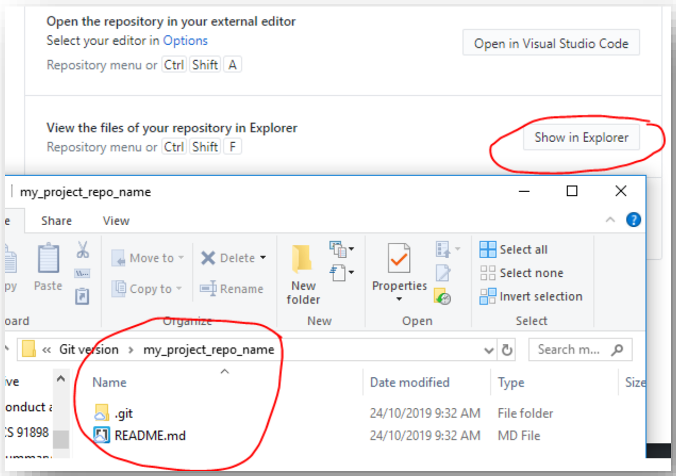
There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Open the repository in your external editor
Select your editor in Options
Repository menu or Ctrl Shift A
Open in Visual Studio Code

View the files of your repository in Explorer
Repository menu or Ctrl Shift F
Show in Explorer

Open the repository page on GitHub in your browser
Repository menu or Ctrl Shift G
View on GitHub

Look in your documents to make sure that you can find your new folder and files. There is also a shortcut to them inside GitHubDesktop, but it’s important to know where they are saved, as you will often be working on them directly. You only really need to use GitHub Desktop when you want to make a Commit and push your files up to the online master repo.


Open the repository in your external editor
Select your editor in Options
Repository menu or Ctrl Shift A
Open in Visual Studio Code

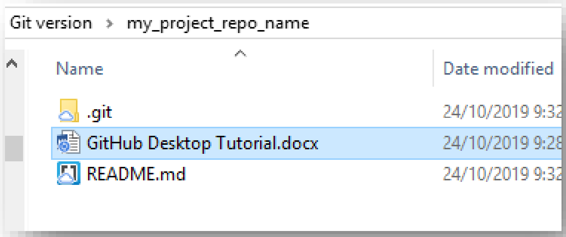
View the files of your repository in Explorer
Repository menu or Ctrl Shift F
Show in Explorer

Name	Date modified	Type
.git	24/10/2019 9:32 AM	File folder
README.md	24/10/2019 9:32 AM	MD File

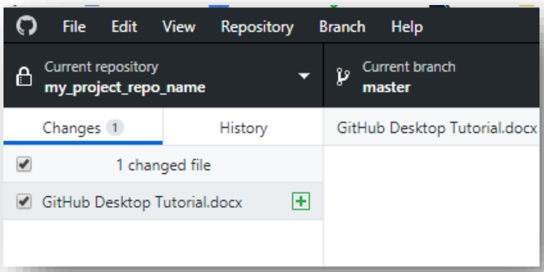
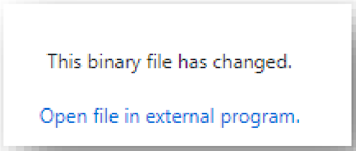
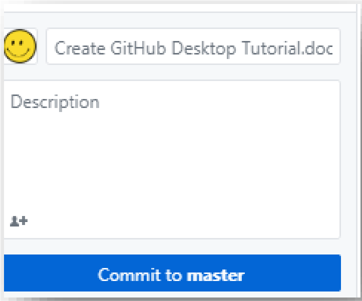
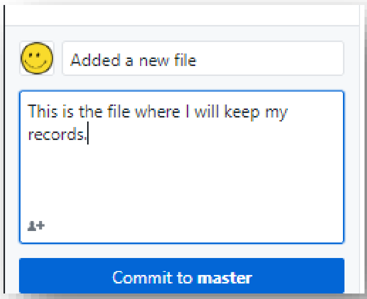
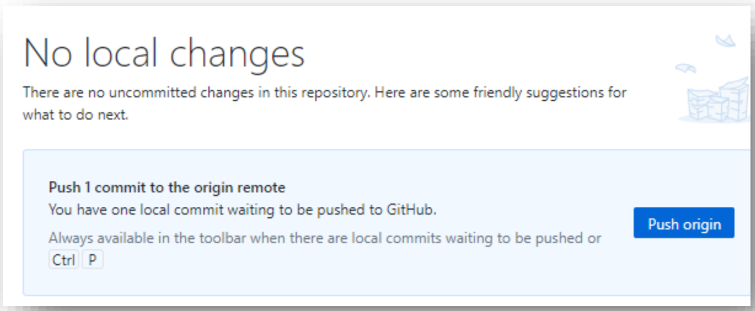
Adding and editing files

Now it’s time to see what happens when you make a change to the files in your local repository, and how to commit these changes and push them back up to your online repo.

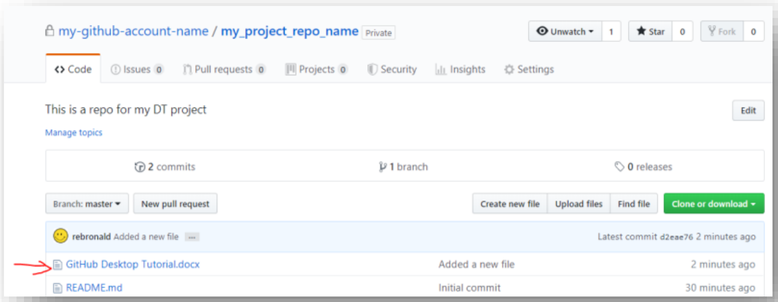
Using your usual editing method, create or paste a new file into your new project folder. How you do this depends on what



Name	Date modified
.git	24/10/2019 9:32
GitHub Desktop Tutorial.docx	24/10/2019 9:28
README.md	24/10/2019 9:32

program and file type you are using – In my screenshot, to keep things simple I've just added a Word document.	
You will see that GitHub Desktop recognises that your files have changed.	 File Edit View Repository Branch Help Current repository my_project_repo_name Current branch master Changes 1 History GitHub Desktop Tutorial.docx 1 changed file GitHub Desktop Tutorial.docx  This binary file has changed. Open file in external program.
Once you have made the changes to your file that you wanted – made edits, added files, you will be ready to make your first 'commit'. You need to add a message and then click the 'Commit it master' button. Sometimes it is tidier if you close the files you are working in first, but this isn't always necessary.	 Create GitHub Desktop Tutorial.doc Description Commit to master  Added a new file This is the file where I will keep my records.
Committing your file doesn't upload your new file to the online origin repo, but it adds it as a commit to your local master version. It tells you that your local master is up to date by saying 'No local changes'. If you choose to, now is a good time to click 'Push origin' and send your latest commit to the online version.	 No local changes There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next. **Push 1 commit to the origin remote** You have one local commit waiting to be pushed to GitHub. Always available in the toolbar when there are local commits waiting to be pushed or Ctrl P

If you check your repo in online GitHub, your changes should now be there. You may need to refresh the page to see them.

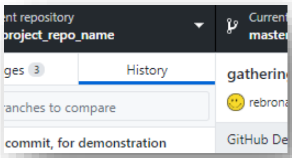


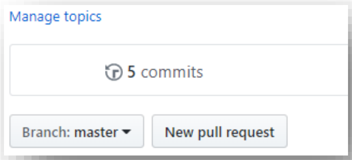
This is a repo for my DT project

rebronald Added a new file		Latest commit d2eae76 2 minutes ago
GitHub Desktop Tutorial.docx	Added a new file	2 minutes ago
README.md	Initial commit	30 minutes ago

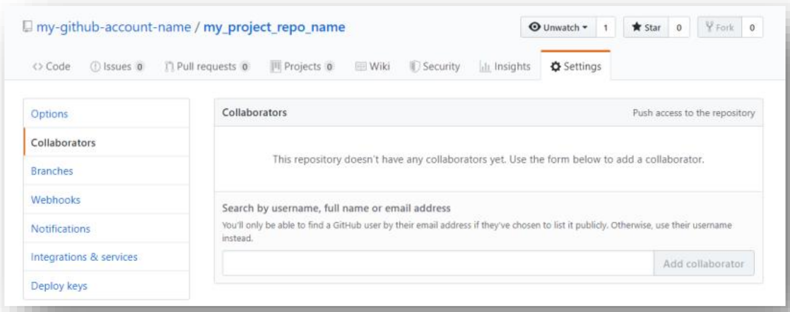
From here on in, your workflow would be that you make changes to your files locally as you develop your outcome and after every significant change (or every few days), commit and push your files to update the online version. Use descriptive commit messages!

If you need to roll back to or view a previous commit, you can click on 'History' in GitHub Desktop, or 'commits' in your online repo.





If you are working with a team, you can add them as 'Collaborators' online. This means you can work on the project at the same time.



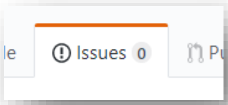
This repository doesn't have any collaborators yet. Use the form below to add a collaborator.

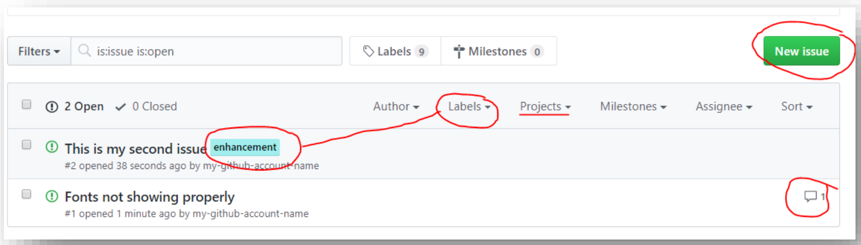
Search by username, full name or email address

You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.

You might want to track issues related to your project or have online discussions with others in your team. Issues is a good place to note things to fix or improve.

Issues lets you categorise items using labels, add issues to your project planning board, and comment on existing issues.





This is my second issue

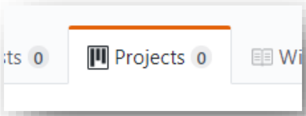
#2 opened 38 seconds ago by my-github-account-name

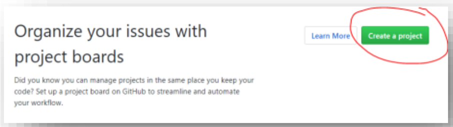
Fonts not showing properly

#1 opened 1 minute ago by my-github-account-name

You can create your project planning or Kanban board within online GitHub. GitHub calls this a ‘Project’.

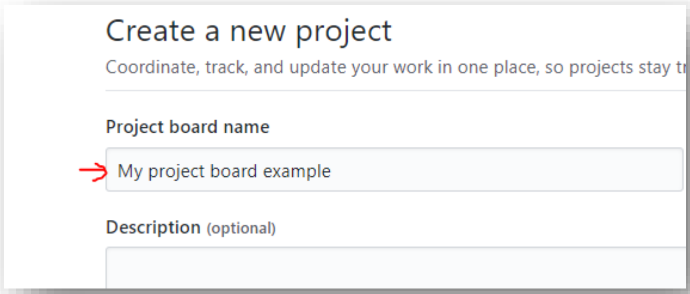
You can add or rename columns, add new tasks and drag to move them across the board. Adding styling with markdown lets you allocate tasks to people on your team, or add checklists and images.





Organize your issues with project boards

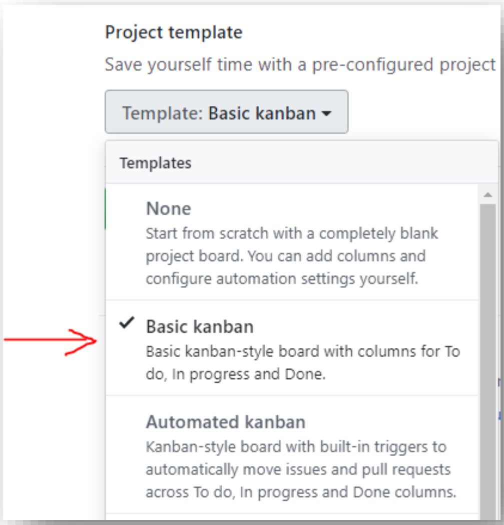
Did you know you can manage projects in the same place you keep your code? Set up a project board on GitHub to streamline and automate your workflow.


Create a new project

Coordinate, track, and update your work in one place, so projects stay tr

Project board name: My project board example

Description (optional)


Project template

Save yourself time with a pre-configured project

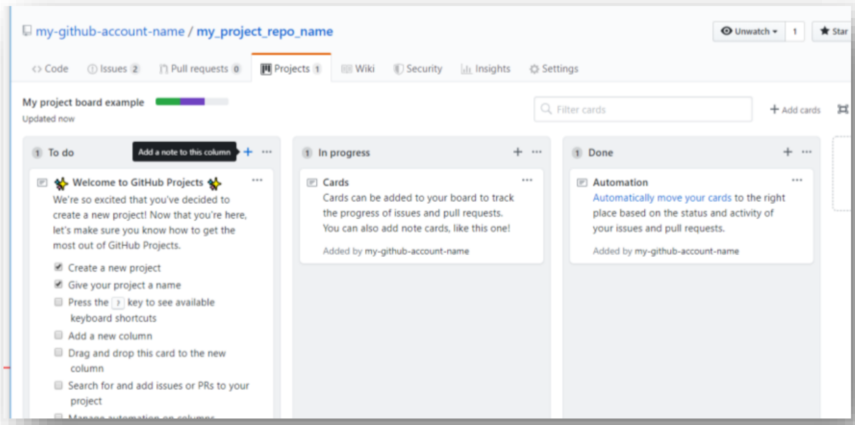
Template: Basic kanban

Templates

None
Start from scratch with a completely blank project board. You can add columns and configure automation settings yourself.

✓ Basic kanban
Basic kanban-style board with columns for To do, In progress and Done.

Automated kanban
Kanban-style board with built-in triggers to automatically move issues and pull requests across To do, In progress and Done columns.



My project board example

Updated now

1 To do

Welcome to GitHub Projects

We're so excited that you've decided to create a new project! Now that you're here, let's make sure you know how to get the most out of GitHub Projects.

- [x] Create a new project
- [x] Give your project a name
- [ ] Press the ? key to see available keyboard shortcuts
- [ ] Add a new column
- [ ] Drag and drop this card to the new column
- [ ] Search for and add issues or PRs to your project
- [ ] Manage automation on columns

1 In progress

Cards

Cards can be added to your board to track the progress of issues and pull requests. You can also add note cards, like this one!

Added by my-github-account-name

1 Done

Automation

Automatically move your cards to the right place based on the status and activity of your issues and pull requests.

Added by my-github-account-name

More tutorials:

- Learn Git with an interactive sandbox at https://learngitbranching.js.org/
- The Git and GitHub for Poets tutorial series at https://www.youtube.com/playlist?list=PLRqwX-V7Uu6ZF9C0YMKuns9sLDzK6zoiV.

- An introductory tutorial lecture is at https://youtu.be/xuB1Id2Wxak - this is a good explanation of version control and Git, and goes on to teaching how to use it and basic commands.
- An alternative tutorial is https://youtu.be/1u2qu-EmIRc - also very long but great information and demonstration.