

Google Colaboratory

A GitHub repository's README .md file is often the first thing potential users or contributors see when they access your repo. It's a crucial document that provides important information about your project.

Motivation

The README .md document often serves as the first point of contact for anyone who stumbles upon your repository. It provides an opportunity to explain what your project does, how to use it, and how to contribute. But beyond this, a well-crafted README can also demonstrate your skills as a developer, your attention to detail, and your ability to communicate complex concepts in a clear and straightforward manner.

Think of it as the front door to your project. It's the first thing potential employers, potential collaborators, or users see when they visit your repository. A good README can draw people into your project, demonstrate your professionalism, and give viewers confidence in your work.

README File Basics

When you create a repository on GitHub, you have the option to initialize it with a README .md file. Once created, GitHub will automatically render the README file and display its content on your repository's main page. This is why your README needs to be impactful - it's the first thing visitors see when they land on your repository.

Initialize this repository with:

☐ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

README files are typically written in the Markdown language. Markdown is a lightweight markup language with a plain-text-formatting syntax. It allows you to easily style text on the web and provides a simple way to control the formatting of a document: setting text to bold or italic, adding images, creating lists, etc. If you aren't familiar with Markdown from elsewhere in the course, you can check out a quick guide [here](#).

The following code snippet shows simple examples of what you can do with Markdown:

```
# This is a Title (H1 heading)
```

```
## This is an H2 heading
```

```
[this is a hyperlink](https://www.google.com)
```

```
- This
```

```
- Is  
- A  
- Bulletpoint  
- List
```

```
1. This
```

```
2. Is
```

```
3. A
```

```
4. Numbered
```

```
5. List
```

```
### This is an H3 Heading
```

This is how you add an image:

```
![image info](/pictures/image.png)
```

You can also use any HTML you want in a Markdown file:

```
<br>
```

```
<p align=center><img src=images/example_image.png width=900></p>
```

```
<br>
```

You can either edit your README file on the remote repo itself on Github, or inside your local copy of your repo using VSCode, and then push the resulting changes to the remote repo.

Sections of a README

Title

The title of your README file should ideally be the same as your project name. Make sure it's descriptive and concise. This will be the first impression users have of your project, so it should quickly convey what your project does.

Description

This section should briefly explain what your project does and why it exists. It could include the problem your project solves, the target users, and any major features. Aim to make this part as compelling as possible, since it's often the first thing people will read. For example if your repository is a piece of software you have written, briefly outline the problem or the use-case that it addresses, or if it's a data project, describe the question you were trying to solve and outline how your project achieves it.

Here's a checklist of items to include in your description. Not all projects will need all of them, but you should at least ask yourself whether each one applies:

What was your motivation? Why did you build this project?

What problem does it solve?

Why did you make the decisions you made in creating your solution?

What did you learn?

What makes your project stand out?

If your project has a lot of different features, consider adding a "Features" section and listing them here

Table of Contents

For larger **README** files, a table of contents is a useful addition. It makes your document easier to navigate. Link each section in the table of contents to the corresponding header in your **README** to allow for quick access.

For example, the following Markdown snippet creates hyperlinks from the entries in the table of contents to their corresponding headings:

```
# Table of Contents
1. [Introduction](#introduction)
2. [Section 1](#section-1)
   - [Subsection 1.1](#subsection-1.1)
   - [Subsection 1.2](#subsection-1.2)

## Introduction
Content here...

## Section 1
Content here...

### Subsection 1.1
Content here...

### Subsection 1.2
Content here...
```

Installation

If your project is a piece of software, like an application or a package that others might download and use, you should provide detailed instructions on how to get your project running on the reader's local

environment. This could include information on necessary software, environment setup, installation of dependencies, and any final steps needed to get the project up and running.

Usage

It is also helpful to give a walkthrough or guide to your project in action. If your repository is a data project, this would be the section where you present your key findings, or it is a piece of software that others might want to use, it should show examples of how to use it. This could include code snippets, command line instructions, screenshots, gifs or videos. The goal here is to demonstrate the value of your project by showing it in action. If your project is a library, then a list of the available functions would go in this section, possibly alongside some worked examples.

Contributing

If you are happy for people to contribute to the project, eg. by suggesting and coding new features, you would include a contribution guide. In this section, outline the process for interested contributors to help improve your project. This could include information on how to file issues, how to propose improvements, and the process for submitting pull requests. You could also mention any specific coding standards or practices you'd like contributors to follow, or any specific features you are looking for that you are hoping others might be able to provide.

License

This part informs users about the license under which your project is released, which defines what they are legally allowed to do with your project. If you're unsure what license to use, platforms like [Choose a License](#) can provide guidance.

How a Good README Can Help Your Career

A well-written **README** is a showcase of your technical skills, attention to detail, and your ability to communicate complex concepts. It demonstrates your writing skills and your ability to explain and document your code, which is a highly sought-after skill in the industry, and one which even good coders often lack.

A well-documented project can also attract more users and contributors, increasing the visibility and impact of your work, which in turn is appealing to employers. Having a project with a thorough **README** on GitHub shows potential employers that you can manage a project, contribute to open source, and work collaboratively: all important skills in a professional setting.

Enhancing Your README File

Badges: Badges are small, dynamic status messages that you can embed in your **README** to provide quick information about the project. For example, build status, number of people following the repository, and latest package version. Tools like [Shields.io](#) offer a wide variety of badges you can use. You can also find

Shields.io's repo [here](#).

Examples

- code coverage percentage:
- stable release version:
- package manager release:
- status of third-party dependencies:
- static code analysis grade:
- SemVer version observance:
- amount of Liberapay donations per week:
- Python package downloads:
- Chrome Web Store extension rating:
- Uptime Robot percentage:

GIFs: Animated GIFs can be an engaging way to demonstrate the functionality of your project. Tools like [LICEcap](#) for Windows and macOS or [Peek](#) for Linux make it easy to record your screen and save the result as a GIF. [GIPHY Capture](#) is another tool for macOS that allows you to create high-quality GIFs.

Table of Contents: A table of contents helps readers navigate your README, especially for larger projects. Tools like [DocToc](#) can automatically generate a table of contents for you.

Screenshots: Including screenshots in your README can give readers a quick understanding of what your project does. Use your operating system's built-in screenshot tools (like **Snipping Tool** or **Snip & Sketch** on Windows, or **SHIFT+CMD+4** on MacOS), then you can annotate and upload these screenshots directly to GitHub when editing your README.

Code formatting and highlighting: Markdown supports inline code and code blocks, but if you want more control over the display and highlighting of your code, you might use something like [Carbon](#). This allows you to create and share beautiful images of your source code.

Charts and diagrams: If your project has a complex architecture, it might be helpful to include a chart or diagram. Tools like [draw.io](#) or [Mermaid](#) allow you to create flowcharts, sequence diagrams, Gantt charts, and more.

Your Github Profile Page

Creating an impactful and engaging GitHub profile README is a crucial part of your online presence, particularly if you're interested in open source development or if you're seeking employment in the tech industry.

How to Create a Profile README

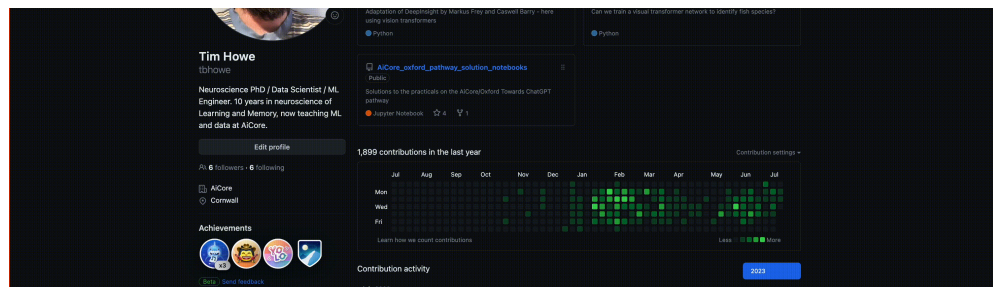
By default, your GitHub profile looks a bit like this:

The screenshot shows a GitHub profile for user 'tbowe'. It includes a profile picture, a bio (Neuroscience PhD / Data Scientist / ML Engineer), and a list of pinned repositories: 'Louis-Wilson-Template-Correlation-Matlab', 'facebook-marketplaces-recommendation-ranking-system', 'deepinsight_transformer', and 'fish_ID_VIT'. Below the profile is a contribution graph showing 1,899 contributions in the last year. The 'Contribution activity' section shows a bar chart for July 2023, indicating 51 commits in 4 repositories.

There's a place for a short biography and a profile picture, and a board where you can pin your favourite repositories, but in general there is very limited room for customisation. Thankfully it's very easy to create a bespoke profile README and display that instead.

- 1. Create a new repository:** Log in to your GitHub account, and click on the + icon on the top right corner next to your profile picture, then select "New repository"
- 2. Name the repository:** The repository needs to have the exact same name as your GitHub username, including capitalization. For example, if your username is "exampleUser", the repository name should be "exampleUser". GitHub will identify this specially-named repository as the place to find your profile README.
- 3. Initialize the repository:** Check the box that says **Add a README file**. This will create a README file in your repository where you can write the contents of your profile README.
- 4. Edit the README:** Click on the README.md file, then click on the pencil icon to edit the file. Here you can add the content you want to show on your profile README.
- 5. Save your changes:** Once you're done editing, scroll down to the bottom of the page, and click the **Commit changes** button

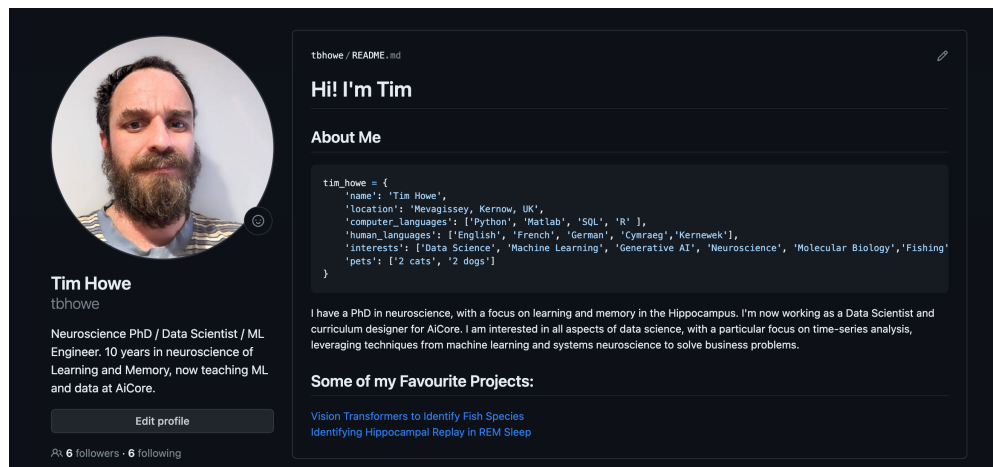
The screenshot shows the 'New repository' dialog box in GitHub. It includes options to 'New repository', 'Import repository', 'New codespace', 'New gist', 'New organization', and 'New project'. The 'New repository' option is selected.



You can then clone this repository to your local machine and edit it like any other repo. I edited mine very briefly as follows:

```
1 # Hi! I'm Tim
2
3 ## About Me
4
5 ```python
6 tim_howe = {
7     'name': 'Tim Howe',
8     'location': 'Mevagissey, Kernow, UK',
9     'computer_languages': ['Python', 'Matlab', 'SQL', 'R' ],
10    'human_languages': ['English', 'French', 'German', 'Cymraeg', 'Kernewek'],
11    'interests': ['Data Science', 'Machine Learning', 'Generative AI', 'Neuroscience', 'Molecular Biology', 'Fishing', 'Electronic Music'],
12    'pets': ['2 cats', '2 dogs']
13 }
14 ```
15
16 I have a PhD in neuroscience, with a focus on learning and memory in the Hippocampus. I'm now working as a Data Scientist and curriculum designer for AICore. I am interested in all aspects of data science, with a particular focus on time-series analysis, leveraging techniques from machine learning and systems neuroscience to solve business problems.
17
18 ## Some of my Favourite Projects:
19
20 [Vision Transformers to Identify Fish Species](https://github.com/tbhowe/fish_ID_VIT)
21 <br>
22 [Identifying Hippocampal Replay in REM Sleep](https://github.com/tbhowe/Louis-Wilson-Template-Correlation-Matlab)
23 <br>
24
```

Once you are happy with what you've written, you can commit and push it to the remote repository and it will display at the top of your profile:



What to Put in your Profile README

Biography

Write a short paragraph or bullet points, no more than around 8 to 10 lines, describing your skills and achievements. Make sure to keep it professional.

Mention your Languages, Packages and Skills

You should make sure to list all the coding languages in which you are proficient, as well as any packages or libraries you use regularly. This is also a good place to mention any relevant non-coding skills, such as Bayesian Statistics or Quantum Physics.

You can add some icons for the languages and libraries you use, with this [repo](#)

There is a wide selection of badges for all kinds of tech-related things [here](#)

Your Github Stats

You can add a summary of your Github stats using this [repo](#)

This gives a quick, nicely-presented overview of your activity on Github, showing how active a developer you are!

Helpful Additional Items

Display your most recent Medium Article on your profile README: [repo](#)

Dynamically generated images from your LinkedIn profile : [repo](#)

Examples of Great README files

Repository README files

The requests python library: [link](#)

VSCode, as an example of a large project: [link](#)

A curated list of projects with awesome README files: [link](#)

Profile README files

You can find a repository containing a curated list of high quality profile README files [here](#)

Your repo's README file serves as the introductory guide for your repository, showcasing the project and reflecting the developer's skills, professionalism, and communication ability

Your repo's **README** is an important tool in showcasing your work and helping you get hired

A project's **README** should contain a descriptive title, compelling project description, a navigable table of contents for larger documents, detailed installation instructions, a usage guide with examples, a contribution guide, and information on the project's license

Your **profile README** should include a short professional biography, a list of your coding languages, packages, and tech-related skills