Data Use Academy and Consulting (DUAC)

DUAC · FREE GUIDE

Build Your Data Portfolio on GitHub

A step-by-step guide to putting your data projects online with Git and GitHub, so an employer can see your work, not just read about it.

Portfolio · Git and GitHub

What you will build

1 · PART 2

A profile page

Your portfolio's front door at `github.com/yourname`: who you are, your skills, and links to your best work.

2 · PARTS 3 TO 5

A home for each project

One repository per project, holding the files and a short case study that explains your thinking.

3 · PART 7, OPTIONAL

A free website

The same work as a web page at `yourname.github.io`, made with GitHub Pages.

Git or GitHub?

Git is a free tool that saves snapshots of your project folder, so you can see what changed and go back if something breaks. **GitHub** is a website that stores those snapshots online and shows them to others. You can do everything in Parts 1 to 5 in your web browser, without installing anything.

How this guide works

Take one project through every part before you start the next. One finished, well-explained project beats five half-finished ones.

ROUTE A · START HERE

Everything in the browser

Parts 1 to 5. Create your profile and first project by clicking and uploading on github.com. Nothing to install.

ROUTE B · WHEN YOU ARE READY

Git on your computer

Part 6, a one-page cheat sheet. Keep projects up to date from your own folder with a few Git commands.

Part 1 Set up your GitHub account

Your username becomes part of every link you share, so choose it with care.

1. Go to `github.com/signup` and sign up with an email address you will keep for years.
2. Choose a professional username, such as `ada-okafor` or `adaokafor-data`. Avoid nicknames and birth years: an employer will read it.
3. Verify your email address, then turn on two-factor authentication when GitHub asks. An authenticator app on your phone is the easiest option.
4. Select your profile picture (top right), then **Your profile** › **Edit profile**. Add a clear photo, your full name, a one-line bio such as *Data analyst · Excel, SQL, Power BI*, your city, and your LinkedIn link.

WATCH OUT · Do not use a school or work email

A student or office email stops working when you leave, and you can lose access to your account with it. Use a personal email you control.

CHECKPOINT · Before you move on

You can sign in, and your profile shows your photo, name and bio.

Part 2 Create your profile page

GitHub shows a special README file at the top of your profile. It works as your portfolio's home page.

1. Select **+** (top right), then **New repository**.
2. In **Repository name**, type your username exactly. GitHub replies that this is a special repository.
3. Choose **Public**, tick **Add a README file**, then select **Create repository**.
4. Open `README.md`, select the pencil icon to edit it, delete the starter text and paste the template on the next page.
5. Replace everything in [square brackets] with your own details, then select **Commit changes** and **Commit changes** again in the window that opens.

TIP · What is a repository?

A repository, or repo, is a project folder that GitHub tracks. Every project in your portfolio gets its own repository, and this special one holds your profile page.

Profile page template

Copy this into your `README.md`. The symbols are Markdown, a simple way to format text: the table below explains them.

```
# Hi, I'm [Your Name]

I'm a data analyst in [City] who turns messy data into clear answers.
I use **Excel, SQL and Power BI**, and I'm currently learning [skill].

## Projects

- **[[Project title]](link to the project)**: [one line on the question
  it answers and what you found].
- **[[Project title]](link to the project)**: [one line].
- **[[Project title]](link to the project)**: [one line].

## Skills

Data cleaning · SQL queries · Power BI dashboards · Excel · Reporting

## Contact

[LinkedIn](https://www.linkedin.com/in/yourname) · your.email@example.com
```

Markdown at a glance

You type	You get	Use it for
`# Title`	**Title**	One main heading at the top. ## gives a smaller heading.
`**Power BI**`	**Power BI**	Bold, for tools and key numbers
`- First point`	• First point	A bullet list, one line each
`[LinkedIn](https://...)`	LinkedIn	A link: words in square brackets, address in round brackets
`![Chart](images/chart.png)`	The picture itself	A screenshot of your dashboard or chart
`| A | B |` `|---|---|`	A small table	Key figures, such as before and after cleaning

Leave an empty line before every heading, list and table, as the template does. Without it, Markdown runs them into the line above.

CHECKPOINT · Before you move on

Open `github.com/yourname`. Your new profile page appears above your repositories. Fill in the project links after Part 3.

Part 3 Create your first project repository

Each project gets its own repository. Give it a clear name, because the name appears in the link you share.

1. Select **+** › **New repository**.
2. Name it with lowercase words and hyphens, such as `delivery-delays-analysis`. Avoid names like `project1` or `my-powerbi-work`.
3. Add a one-line **Description**, such as *Which locations cause late deliveries? Excel, SQL and Power BI.*
4. Choose **Public**, tick **Add a README file**, then select **Create repository**.
5. Select **Add file** › **Upload files**, drag in your files, write a short message such as *Add cleaned data and dashboard*, then select **Commit changes**.

A tidy folder layout

Employers open your repository and look for the right file in seconds. Use the same layout for every project.

File or folder	What goes in it
`README.md`	The case study that explains the project (Part 4). This is what people read first.
`data/`	The data, or a small sample of it, only if you are allowed to share it. Otherwise, a link to where it comes from.
`analysis/`	Your SQL queries, Excel workbook, R or Python scripts.
`dashboard/`	The Power BI file (`.pbix`) or a PDF export of the report.
`images/`	Screenshots of the dashboard and key charts, to show in the README.

To create a folder in the browser, select **Add file** › **Create new file** and type a name ending in a slash, such as `images/`, then a file name. Or name your files with the folder in front when you upload them from your computer through Route B.

WATCH OUT · Never upload private data

No workplace data, customer, patient or staff records, or anything you do not have permission to share. A public repository can be read and copied by anyone. Use public, fictional or properly permitted data, and say where it came from.

TIP · File too big?

The browser takes files up to 25 MB, and GitHub refuses anything over 100 MB. For a large Power BI file, upload screenshots and a PDF export, and link to a published report instead.

Part 4 Write the project's case study

The README turns files into evidence. Edit the project's `README.md` with the pencil icon, paste this outline, and answer each heading in a few plain sentences.

```
# Delivery Performance Across Five Locations

## Background        Why this question matters, in two sentences.
## Objective         The question or decision the analysis supports.
## Data              Source, period covered, number of rows, limitations.
## Process           How you cleaned, analysed and checked the data.
## Findings          Three to five results, each with a number.
## Recommendations   Next steps the evidence supports. No more.
## Tools used        Each tool and what you used it for.

![Late deliveries by location](images/late-by-location.png)
```

TIP · Write a finding, not a caption

"Lagos had the most late deliveries" describes a chart. "Lagos had the most late deliveries, but also the most orders, so the next step is to compare delay rates" shows how you think.

WATCH OUT · Picture not showing?

The path in brackets must match the file exactly, including capital letters: `Chart.png` and `chart.png` are different files. Avoid spaces in file names.

Part 5 Make your portfolio easy to find

1. On each project, select the gear icon next to **About**. Check the description and add topics such as `power-bi`, `sql` and `data-analysis`.
2. On your profile, select **Customize your pins** and pin your best projects. You can pin up to six; three strong ones is plenty.
3. Go back to your profile README (Part 2) and replace each *link to the project* with the project's real address.
4. Add `github.com/yourname` to your CV, your LinkedIn **Featured** section, and your email signature.

CHECKPOINT · Route A done

Your profile page links to at least one project, and that project has its files, a case study and a screenshot. You have a portfolio. Part 6 shows how to keep it up to date from your own computer.

Part 6 Route B: Git on your computer

Once a project has more than a few files, it is easier to work in a folder on your computer and send changes to GitHub with Git. Set up once, then repeat the everyday loop each time you work on the project.

Set up once

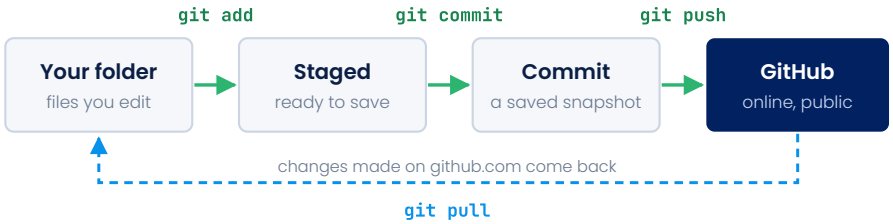
1. Install Git from `git-scm.com/downloads`, keeping the default options.
2. Open **Git Bash** (Windows) or **Terminal** (Mac), and tell Git who you are:

```
git config --global user.name "Ada Okafor"
git config --global user.email "ada.okafor@example.com"
```

3. Copy the project's address from the green **Code** button on GitHub, then download it:

```
git clone https://github.com/ada-okafor/delivery-delays-analysis.git
cd delivery-delays-analysis
```

How a change travels



TIP · The first push asks you to sign in

A browser window opens: sign in to GitHub there. Your GitHub password does not work typed into Git Bash. Prefer buttons to commands? **GitHub Desktop** (`desktop.github.com`) does the same steps.

The commands you need

Command	What it does	When you use it
`git clone [address]`	Downloads a repository as a folder	Once per project
`git status`	Lists what has changed	Any time you are unsure
`git add .`	Stages every change in the folder	Before each commit
`git commit -m "..."`	Saves a snapshot with a message	After each finished piece of work
`git push`	Sends your commits to GitHub	When you want it online
`git pull`	Brings in changes made on GitHub	Before you start working
`git log --oneline`	Lists your past commits	To see the project's history

The everyday loop

```
# 1. Start by bringing in anything changed on GitHub
git pull
# 2. Work on your files as normal: Excel, Power BI, SQL
# 3. Check, stage, save and send
git status
git add .
git commit -m "Add delay rate by location to the dashboard"
git push
```

Part 7 Publish a free website (optional)

GitHub Pages turns a repository into a web page. It is optional: your GitHub profile already works as a portfolio.

1. Create a repository named `yourname.github.io`, using your username, with a README. Put your profile text from Part 2 in it.
2. Open the repository's **Settings** › **Pages**.
3. Under **Build and deployment**, set **Source** to **Deploy from a branch**, choose **main** and **/ (root)**, then select **Save**.
4. Wait a few minutes, then open `https://yourname.github.io`. Your README now shows as a web page.

You can do the same on any project repository. Its page appears at `https://yourname.github.io/project-name/`.

TIP · Give it a design in one line

Add a file named `_config.yml` containing `theme: jekyll-theme-cayman`, commit it, and wait a few minutes. GitHub lists its other free themes on its Pages documentation.

Part 8 Check before you share

Open your profile in a private browser window, as an employer would see it, and tick each line.

Check	Done
Your profile page shows your name, a one-line summary and your contact details.	☐
At least one project is pinned, and its name says what it is about.	☐
Each project's case study says the problem, the data, the process, the findings and the limitations.	☐
Every screenshot shows, and every link opens, in a private window.	☐
Nothing confidential or personal is in any repository, including old commits.	☐
Your GitHub link is on your CV and LinkedIn profile.	☐

CHECKPOINT · You are ready to share

Ask someone you trust to open your link and tell you, in one sentence, what your best project found. If they can, your portfolio is doing its job.

Common problems and fixes

What you see	Likely cause	Fix
Your profile README does not show	The repository name does not match your username exactly, or it is private	Rename it to your exact username and set it to Public in **Settings**
"File too big" when uploading	The browser takes files up to 25 MB	Use Git (Part 6) for files up to 100 MB, or share screenshots and a PDF export
A picture in the README is broken	The path or capital letters do not match the file	Copy the file name exactly, and avoid spaces in it
Headings or lists run together	No blank line before them	Leave an empty line before every heading, list and table
"Author identity unknown"	Git does not know your name yet	Run the two `git config` lines in Part 6
"not a git repository"	Git Bash is open in the wrong folder	Use `cd` to move into the project folder, then try again
Push refused: "fetch first"	Something changed on GitHub that your computer does not have	Run `git pull`, then `git push` again
Sign-in or password refused	GitHub does not accept your account password in Git	Sign in through the browser window that opens, or use GitHub Desktop
The Pages site shows "404"	It is still building, or Pages is not switched on	Wait a few minutes, then check **Settings** › **Pages**
You uploaded private data by mistake	Deleting the file leaves it in the repository's history	Make the repository private at once, then delete it and start a clean one

TIP · Keep building

A portfolio grows one project at a time, so add a new one as you learn and improve the older ones as your skills grow. For help choosing projects and writing up what you found, read **No Experience Yet? Build Your First Data Analytics Portfolio This Way** on the DUAC blog at `du-ac.com/blog`.