

# Publishing data using Github

This guide has been created to help you take advantage of Github features for publishing data. Github is not a dedicated data publishing platform, however it is a very powerful, capable platform on which to publish high quality data.

In order to get the most from this exercise you will need a Github account that has web hosting (gh-pages) enabled. To check that your account is setup correctly, there is a complementary guide, “Getting started with web hosting in GitHub”.

## Step 1 - Make a copy of the template repository

The quickest way to get started with a project in Github is to clone (or fork) an existing project as a template. Publishing data is no exception. The Open Data Institute has a template for publishing data in Github.

*<https://github.com/theodi/data-publishing-template>*

Enter this URL in your web browser and we will start by forking the repository. Ensure you are logged in and then click the fork button towards the top-right corner of the Github webpage.



This will create your own copy of the project, in your own workspace, that you can edit. Note that the URL in your browser will now contain your username rather than *theodi*.

## Step 2 - Rename the project

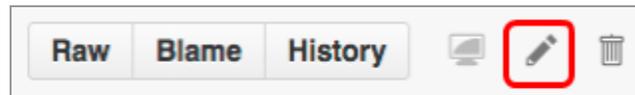


“data-publishing-template” is not a very good name for our project. To rename the project click the **settings icon** (on the right hand side) of your project.

At the top of the setting screen you will find an option to rename the project. Change the name to something that represents the title of the dataset you are publishing. Once done click **rename**.

### Step 3 - Add dataset metadata

Throughout this exercise we will be editing a number of files in Github. One of the main files we need to edit is `_config.yml`. This file can be found at the top level of our project. Open it by clicking on its name. On the following screen you will be able to view its contents and edit it by clicking the **edit icon**.



Note also the delete (trash) icon that we will be needing later to clean up the template.

Edit this file to fill in all the data you possibly can, enter this after the colon “:” for each field. If you do not know what to fill in then remove the text after the colon.

Once done scroll to the bottom of the page and click commit changes. Feel free to add a commit message so that people can easily find what you changed and when.

**Commit changes**

Update `_config.yml`

Added dataset title, description, publisher and license information. Left certificate data blank for now.

### Step 4 - View the dataset webpage and data

Now we have made a change, Github should generate the web pages associated with the project. To access the web page click the settings icon (see step 2), scroll down on the settings page to find the **Github pages** section. This section should list the location of your website corresponding to this project, it may take a few minutes for this to update.

**GitHub Pages**

✓ Your site is published at <http://davetaz-test.github.io/dr-who-episodes-dataset>.

**Update your site**

Changes may take up to **ten minutes** to be visible.

To update your site, push your HTML or [Jekyll](#) updates to your `gh-pages` branch. [More info](#).

Open this webpage in a new tab or window of your browser. As we make each change these will be reflected on this page.

## Step 5 - Upload the dataset

To add data to our dataset upload the files into the **data** directory; select from the top level view of the project (back from Settings).

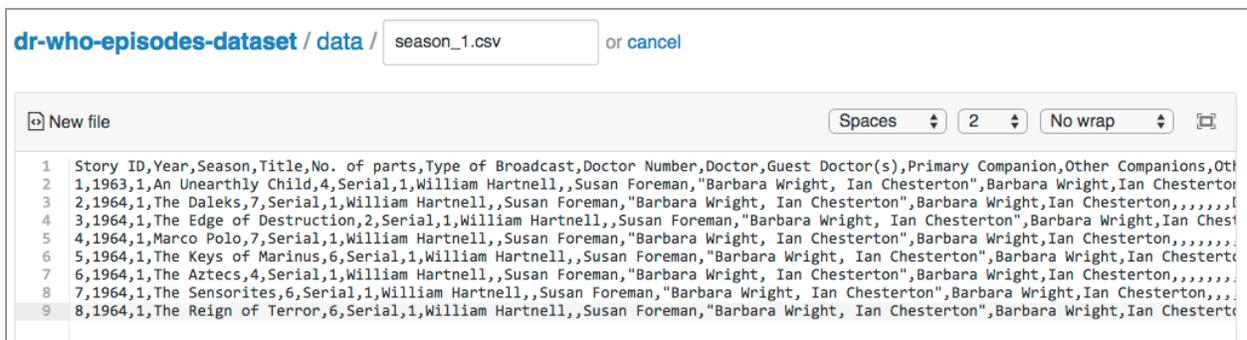
Notice that the data directory already contains a number of files. We will remove these files later, but for now they act as good templates to copy.

To upload new data to this directory click the plus near the top of the screen.



This will create you an edit interface where you can name the file and paste in your data.

Note: you cannot paste a csv from an Excel file, use a notepad type program from which to copy the csv data.



**Note:** There are many ways to manage github projects that allow mass uploading of data from a directory on your computer. This would avoid the need to copy and paste data. This guide uses the Github web interface so that extra software beyond a web browser is not required for this exercise.

Once you have pasted in your data and added a commit message, commit the changes.

## Step 6 - Describe the data file

In order to display each data file in the web interface we also need to describe the file. This is done by adding a *filename.data* file to the data directory. Note that the prefix of the filename **must** match that of your data file. e.g. *my\_data.csv* **must have** a corresponding *my\_data.data* file. Click the **add button** to create a new file following this template.

The contents of this file should look something like the following, note that the dashes at the beginning and end are important and must remain.

```
---  
category: data  
filename: season_1.csv  
weight: 1  
title: Season 1  
description: Episodes in Dr Who season 1  
type: text/csv  
---
```

**Do not** edit the category, however all other values can be changed as follows:

filename: The name of the corresponding data file.

weight: The position the file should appear (1 = 1st, 2 = 2nd and so on)

title: A title for the file

description: A description of the file's data

type: The IANA mime type of the file (text/csv = csv, application/json = json)

When you have completed editing this file, once again click commit.

At this point you should be able to refresh the web interface a few times to see the new file appear. If it doesn't then check that your *.data* file looks like one of the templates.

## Step 7 - Tidying up

**1)** Once you have successfully published your first data file, repeat steps 5 and 6 to upload the rest of the data. At this point you can also delete the example data. This can be done by opening the file we no longer need (as shown in step 3) and clicking the delete (trash) button for each file. Note that you will need to commit a delete.

**2)** As you forked a repository you will need to re-enable issue tracking to allow people to comment openly on errors in the data. Do this from the settings page of the repository.

Note that for the webpage to update you will need to make a change.

## Step 8 - Create an open data certificate

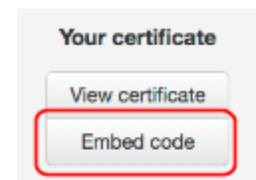
Once you have uploaded your dataset, the last stage involves creating an open data certificate for your dataset at <https://certificates.theodi.org>

To start your certificate, enter the web address where you have been previewing your changes. You should find that the certificate is able to auto-fill a number of fields. This is because the dataset publishing template we used in this guide takes the values in the config files and embeds them in your web page.

The data publishing template also generates a *datapackage.json* file at the top level of your website that conforms to the Open Knowledge tabular data packaging format (more details at <http://dataprotocols.org/tabular-data-package/>)

Once you have completed your certificate and clicked published you will be able to embed the certificate into your website.

This can be done by clicking the **embed code** button from your certificates dashboard. You can then simply copy and paste the **badge** code into the `_config.xml` file from step 3.



## Extension - Customise your website (Advanced)

What we have created in Github is a complete stand alone website. You can not only customise the data, but also the look and feel of the website. For example you could upload a new website logo into the images directory (requires software).

Alternatively you could edit the html templates in the `_includes` to change the structure of the website. To change the style (e.g. colours), you can also edit the `style.css` file in the css directory.