
1 Extra Credit Assignment Introduction

This is an optional non-coding extra credit assignment, and its score will be added to your final assignment grade. That is, if your assignment grade average is 89, and you receive 7 points in this assignment, then your final assignment grade will be $89 + 7 = 96$. You are expected to finish this assignment before 11:59pm on the listed date. Unlike previous assignments, **late submissions will NOT be accepted for this assignment under any circumstances**. Also, there is no flash quiz associated with this assignment. This assignment will be graded after the final exams are graded.

In this assignment, you will take the first steps towards creating and publishing your personal website, which can potentially be turned into your professional online portfolio to bolster your resume. To do this, we will be taking advantage of GitHub Pages; a free website hosting service (with automated deployment) offered by GitHub.

Good luck!

2 Creating a website with GitHub Pages (2 points)

Introduction:

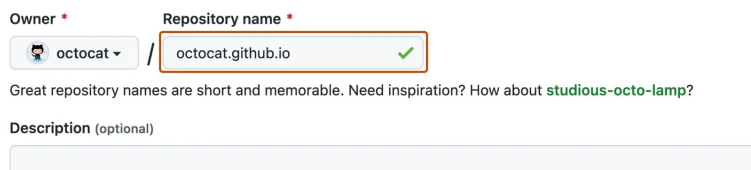
You can host a public website about yourself using GitHub Pages directly from a repository on GitHub. In this part of the assignment, we will be learning how to create and publish a very basic web page through GitHub. You may skip the initial setup below, if you have already done Assignment 5.

Initial setup:

- First, create a free GitHub account on [GitHub.com](https://github.com), preferably using your MSU email.
- Afterwards, complete the ‘Hello World’ introductory tutorial up to Step 1. If you have done this step correctly, you should see your repository under the ‘Dashboard’ on the main page.

Creating a repository for your website:

- Create a new **public** repository with the ‘Add a README file’ textbox checked, and name it `username.github.io`, where `username` should be replaced with your GitHub username. For example:



Owner * Repository name *

octocat / octocat.github.io ✓

Great repository names are short and memorable. Need inspiration? How about studios-octo-lamp?

Description (optional)

- From your command line, **clone** the repository into a local directory. Alternatively, you may use the GitHub Desktop application, if you prefer a graphical user interface. Although this application has limited functionalities, it is still a viable option for managing repositories that are not related to coding. If you choose this option, you should follow the Guide on GitHub Desktop.

Deploying your website:

- Replace the text inside the `README.md` file (which is currently only your repository name) with something else, and save your changes. Then `add`, `commit`, and `push` your changes to your remote repository.
- Your website will now build with the changes you just pushed, and redeploy. You can check the build and deploy status by clicking 'Actions' tab on your repository page.
- A green tick indicates that your website has been successfully deployed. Once you see this, go to `https://username.github.io/`, again replacing `username` with your GitHub username.
- You should now see the changes you made inside the `README.md` file appear on your website. Great success!
- Take a screenshot of the website, and save it as `basic.png`. This is one of your required submission files.

3 Generating a website using Jekyll (2 points)

Introduction:

Although designing your website from scratch would certainly give you the most creative and artistic control over the result, doing so usually comes at the cost of a significant time investment. It is often a viable (and in most cases, better) alternative to use a template as a starting point, and then build on top of it. In this part of the assignment, we will be learning how to apply Jekyll themes to our website.

Initial setup:

- Delete the repository you just created on GitHub. In order to delete your remote repository, go to your repository settings by clicking ‘Settings’ tab on the repository page. Scroll all the way down, and click the ‘Delete this repository’ button. Make sure that you also remove the local copy of the repository.
- Go to github.com/sharu725/online-cv, and click the green ‘Use this template’ button. From the dropdown menu, select ‘Create a new repository’, and just like before, create a **public** repository with the name `username.github.io`, where `username` should be replaced with your GitHub username.

Making changes to your Jekyll theme:

- Once your website is deployed, make sure your website is working properly by going to <https://username.github.io/>, again replacing `username` with your GitHub username.
- Now, `clone` the repository into a local directory.
- Change the name from ‘Alan Doe’ to your name, and then `add`, `commit`, and `push` your changes to your remote repository.
- Once your website is deployed, go to your website, and take a screenshot. Save your screenshot as `default.png`. This is one of your required submission files.

4 Getting started with your website (6 points)

Introduction:

Now that you know how to generate websites using GitHub Pages and Jekyll, you are now ready to design your own website! The Jekyll theme we used earlier was only for demonstration purposes. In this part of the assignment, you will have the complete freedom to design your website in any way you desire.

Initial setup:

- Once again, delete the repository you just created (last time, I promise!). Make sure that you also remove the local copy of the repository.
- Head over to github.com/topics/jekyll-theme. You can find thousands of Jekyll themes listed there, and most Jekyll theme repositories will have a demo website listed under the ‘About’ section that will give you an idea of what the theme will look like. Pick the one that resonates with you the most.

Using your preferred Jekyll theme:

- If your preferred Jekyll theme repository does not have a ‘Use this template’ button, click the ‘Fork’ button instead, and then create the repository like you normally would. Once your forked repository is created, go to ‘Settings’, and then ‘Pages’. Under the ‘Branch’ section, switch to ‘master’ from ‘None’, and hit the ‘Save’ button. Your website should now deploy normally.
- If your build fails, check the usage instructions of the Jekyll theme repository that you are using to make sure that you have not missed anything. If all else fails, it may be the case that the Jekyll theme you have selected has not been updated recently. In that case, just pick another Jekyll theme.
- Once you have ensured that the Jekyll theme you have selected is working properly, `clone` the repository into a local directory, and start designing your new website!

General guidelines:

- Generally, your website should have everything your potential employer should know about you. However, you should also be careful not to include an overwhelming amount of information. One way to achieve this balance is by creating multiple tabs for different categories, and adding only relevant information under said categories.
- Keep in mind that this is not a programming assignment that you can ‘complete’ in the conventional sense. So please refrain from asking us what the minimum amount of work you need to do to get full points here. We will be evaluating your website based on how much effort went into its design, which will be evident from your git commit history. Your website should not be just a Jekyll theme with some changes to the texts. There are plenty of resources online for you to draw inspiration from.
- Start early! Troubleshooting any issues that you encounter is a part of this assignment. There are thousands of tutorials and guides available online to help you with this. If you run into any issues with the Jekyll theme you have chosen, you will need to allow yourself enough time to either troubleshoot the issue, or find an alternative Jekyll theme.

5 Assignment Submission Process

Introduction:

Like all the other assignments in this course, this assignment has a certain file and folder structure that your submission file must adhere to. However, since this is an optional extra credit assignment, unlike previous assignments, **if your file and folder structure are incorrect, you will receive a zero**. Any student who is still not able to follow the submission process properly is not eligible for these extra credit points. Therefore, it is in your best interest to read the instructions listed below thoroughly several times.

Submitting your assignment:

- Create a folder, name it `your_msu_idec`. For example, if your MSU email is `johndoe@msu.edu`, then you should name the folder `johndoeec`.
- For each task, create a sub-folder inside your `your_msu_idec` folder, and name it as the number that corresponds to the task number. For this assignment, there should be three sub-folders named '2', '3', and '4'.
- Inside sub-folder '2', put **ONLY** the `basic.png` image file.
- Inside sub-folder '3', put **ONLY** the `default.png` image file.
- Inside sub-folder '4', create a text file called `git.txt` that contains **ONLY** the HTTPS URL of your repository. Your HTTPS URL is **NOT** your website or repository link. If your `git.txt` file contains anything other than the correct HTTPS URL that ends with `.git`, you will not get credit. In order to obtain your HTTPS URL, go to your repository page, click the green <> `code` button, and select the HTTPS option. Then copy the HTTPS URL and paste it into your `git.txt` text file.
- Compress/Zip your `your_msu_idec` folder and name it `your_msu_idec.zip`. For example, if the name of your folder is `johndoeec`, then you need to create a zip file named `johndoeec.zip`. Guide for creating zip files: copyrightservice.co.uk/reg/creating-zip-files.
- Submit the zip file through D2L.
- We will be evaluating your website based on the commits made up to the time of submission. So if you make any changes after submitting your zip file (but before the deadline), make sure you resubmit the same zip file, so that the timestamp of file submission is updated.