CSC122 Web Project

Assignment:

Using Flask, HTML, CSS and whatever additional resources you need, create a simple one-page "to-do" list that performs CRUD (create, read, update, delete) on a data source. Manage this application with GitHub and deploy the final version to Heroku.

The to-do list can actually be to-do, or it could be a list of anything- shopping items, project items, car collection, store inventory, etc...  make sure it's something meaningful and have 3-5 columns in your list.

When you decide on your “list” submit a description of your list complete with all the columns.

**Prior approval is required before you begin work.**

Guidelines

Create a master folder for the project (I'd recommend a repo in GitHub) and make sure within that folder you have folders for images, scripts, and styles. Create a master stylesheet (i.e. default.css) and use that to format your page.

Your page should have obvious and easy-to-use functionality for adding listing, records, editing those records, and deleting them. It should use ONE page to do all these things- but using code and routing, you can have the page display different contents. Note that many examples and tutorials put some of this functionality on separate pages.

Tutorials

You can get a good start with this tutorial: <https://www.youtube.com/watch?v=Z1RJmh_OqeA>

 and a number of other ones here: <https://hackr.io/tutorials/learn-flask?sort=upvotes&type_tags%5B%5D=1>

There is also a good kickstart for flask and github on harvard's "beyond CS50" class:

- Git and github: <https://www.youtube.com/watch?v=eulnSXkhE7I&t=73s> (I recommend github desktop for windows and visual studio code for coding and github integration - as well as python/flask integration)

- flask (along with some CSS and such): <https://www.youtube.com/watch?v=pP23CtOI-1U>

**Additional Flask websites / tutorials**

* <https://flask.palletsprojects.com/en/1.1.x/>
* <https://flask.palletsprojects.com/en/1.1.x/tutorial/>
* <https://realpython.com/tutorials/flask/>
* <https://pythonspot.com/flask-web-app-with-python/>
* <https://codingcompiler.com/flask-tutorial-create-micro-blogging-application/>

Review/Grading/Checklist

In evaluating your project we will do the following:

* load your URL and see if the application launches from Heroku (or any public website starting with http....)
* Look for your unique colors/design/name on the page and in the html title element.
* Check your HTML against the validator (validator.w3.org)
* Check your CSS against the validator (<https://jigsaw.w3.org/css-validator/>​)
* Look for an obvious way to add an item (and appropriate instructions/explanations/icons)
* Add an item - so now the list shows the additional thing you just added.
* Refresh page to confirm that it stayed there (i.e. was stored in data source)
* Look for obvious way to delete an item
* Delete and confirm deletion
* Add several items and delete a couple and confirm again
* Look for obvious way to update/edit items
* Update/edit several items and confirm that those changes saved
* Test any additional visible interface/links/items (i.e. if it's there, it needs to be clear and to work).

**For Extra Credit include a login field**

Suggested To-Do Lists

* Grocery shopping list
* College dorm moving list
* Car repair list
* Home repair list
* Holiday gift list
* Beach vacation items list