Lab-2 Templates (based on Flask-Demo-Lectures/02-Templates) and Video-01-Flask-HelloWorld and Video 02-Templates

You will end up with three project folders:

02-Templates-Lab-*yourname*

02-Templates-Lab-Lists-*yourname*

02-Templates-Lab-Dict-*yourname*

Part 0: Complete the "Setup" lab, and make sure that you can run flask.

Part 1: Use a simple list for favSnacks https://www.w3schools.com/python/python_lists.asp

Copy the entire folder 02-Templates. Rename it 02-Templates-Lab-*yourname*

Add a route named "lab"

In the def of that route:

Create a list (a python list) of your favorite snacks, named favSnacks

(You will send favSnacks to your template. But in your template, it's "posts")

You will also send a title of *yourname* to your template.

So you have to know how to send the parameters properly

The template you will use for this route is lab.html (see below)

Create a template named lab.html

Model it after home.html--it will inherit from layout.html

Keep the variable "post" in your template (even though it's not posts anymore)

You can test the project at this point. Type in your browser localhost:5000/lab

Because every route defines its own url and page.

In layout.html (because lab.html will extend layout.html):

- Create a link on the navigation bar for the lab page.

  Find where you have a link for the home and about pages. Create a simlar link for lab.

- I'd like the navigation bar to be green. You'll need to look up the bootstrap class for this.

  see the background classes:

  https://www.w3schools.com/bootstrap4/bootstrap_colors.asp

- Also, make the link for "home" show up in bright blue. It doesn't look good; do it anyway.

Your lab.html template will accept the title and the favSnacks (which it calls posts, since we didn't change it).

The template will display:

Here are some of *yourname* 's snacks:

[list the snacks here.]

Unlike the original home.html posts, your snacks are just strings.

So you don't have something like post.title or post.author.

Your "post" doesn't have any attributes that you'd need to use with dot notation.

Part 2: Use a list of lists for favSnack, because you want to store the calories too.

Copy 02-Templates-Lab-yourname, and rename the copy to 02-Templates-Lab-Lists-yourname

In your lab route:

favSnacks=[["Nestles's Crunch", 250],["Lay's All-Natural Potato Chips with Sea Salt",200],["Something Healthy",150]]

In your lab.template:

For every post, display the whole post, no changes from the first time.

Note that it displays the list, as a list.

Now change it to display post[0], and then post[1]

displays nicely.

Change post to favSnack. It's too confusing to call it a post.

Be sure to also change post to favSnack in your lab route in flashDemo.py too!

Part 3: Use a list of dictionaries so that your template can use the key names instead of [0] or [1].

Copy 02-Templates-Lab-Lists-*yourname* , and rename the copy to 02-Templates-Lab-Dict-*yourname*

Try this as a regular python program, or even in idle shell:

```
Snacks = list()
Snack = dict()
Snack['snack']="Nestle's Crunch"
Snack['calories'] = 101
Snacks.append(Snack)
print(Snacks)
Snack['snack']="KitKat"
Snack['calories'] = 202
Snacks.append(Snack)
print(Snacks)
```

Notice that Snacks now has two identical entries! What happened to the first snack??!!

```
# When you add something to the list, the list contains a reference to that object, not a copy!!
# So, if you change the value of the Snack dictionary entry (which we are doing), and append it again,
# you will end up with two references to the the updated dictionary entry.
# The solution is to create a new dictionary for each entry to be added to the list.
Snacks = list()
Snack = dict()
Snack['snack']="Nestle's Crunch"
Snack['calories'] = 101
Snacks.append(Snack)
Snack=dict()
print(Snacks)
Snack['snack']="KitKat"
Snack['calories'] = 202
Snacks.append(Snack)
print(Snacks)
```

If you try to do this in your flaskDemo.py program (lab route), you have to redeclare the dict() for each snack.

In your lab.template, favSnack[0] is the same thing as favSnack['snack']. Try it both ways.