CKCS 145: Lab 2.1 - Application Development and Version Control

In this lab, you will be working with Git, Python, Flask and the command prompt. You will need to use the virtual machine (VM) that was set up in Lab 1. To start your VM, log into the machine using your account ("TorontoMet" is the username and "torontomet123" is the password). Open up a Terminal or command prompt and start with step 1.

Prerequisites:

- Flask version 3.0.3 or higher
- Werkzeug 3.0.3 or higher

1. In an earlier lab, you created a directory/folder named "lab2". Navigate to this folder.

   ```
   cd lab2
   ```

2. Use pip to list the Python packages available. Is Flask one of the packages available?

   ```
   pip list
   ```

3. If Flask is not available, then it can be installed using pip.

   ```
   pip install flask
   ```

4. Create a text file named "test.py".

   ```
   touch test.py
   ```

5. Open the above text file using a text editor and add the following inside it.

   ```
   from flask import Flask, flash, redirect, render_template, request,
   session, abort

   app = Flask(__name__)


   @app.route('/')

   def test1():

      return 'Accessed endpoint powered by Plask and Python'


   @app.route('/param')
   ```

```
def param_home():
   return 'Parameter may be submitted to this url.'


@app.route('/param/<name>')
def param_submit(name):
   return 'Parameter %s!' % name


if __name__ == '__main__':
   app.run(host='0.0.0.0', port='5090')
```

6. Start the above application.

```
python3 test.py
```

7. The above application is middleware. This means that it is a service that may be accessed on port 5090 using a web browser. The URL that may be used is http://localhost:5090. What do you see in your web browser when you visit this URL?
8. Use a web browser and visit http://localhost:5090/param. What do you see? Then visit http://localhost:5090/param/echo. What do you see?
9. Modify param_submit() method from test.py file to have the following body.

```
return render_template('test.html', name=name)
```

10. Create a directory called “templates”.

```
mkdir templates
```

11. Navigate to the “templates” folder.

```
cd templates
```

12. In “templates” folder, create a file named “test.html”. Add the following code to this file.

```
<html>
    <body>
         <h1>Your parameter is {{ name }} </h1>
          <p>You are using jinja2 to create this webpage.</p>
    </body>
</html>
```

13. Navigate to the “templates” folder.

```
cd templates
```

14. Use a web browser and visit http://localhost:5090/param. What do you see? Then visit http://localhost:5090/param/echo. What do you see?

15. List the tracked and untracked files by Git of the “lab2” directory/folder.

```
git status
```

16. From the “lab2” directory/folder, add to Git files not tracked.

```
git add .
```

17. Note: files are tracked by Git, but changes have not been saved or committed. Commit these changes now using the below command.

```
git commit -m “Added a Flask app that uses a jinja2 template”
```

18. Display the Git history for this repository as was done in Lab 2.0.

```
git status
git log
```