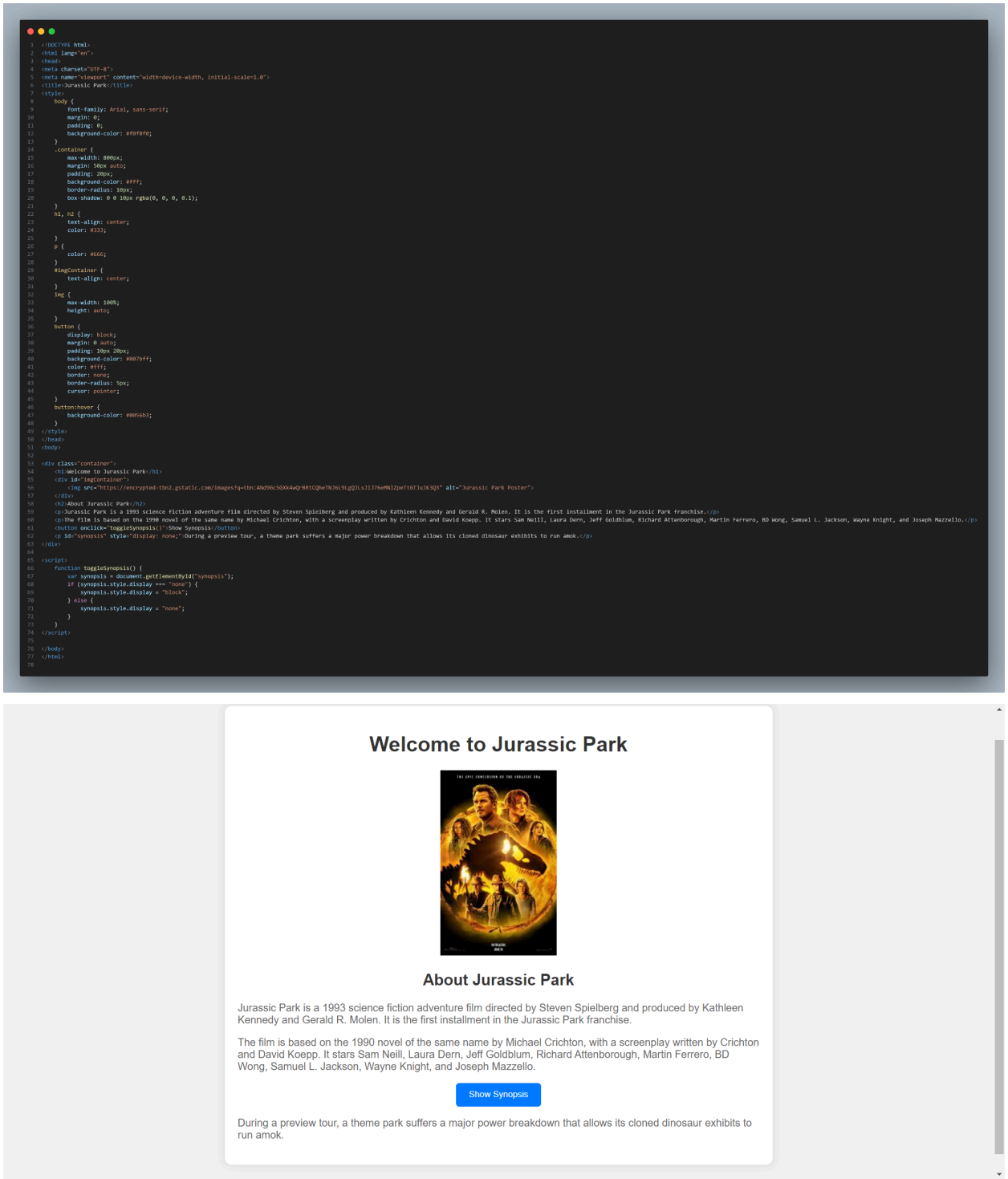
Basig, John Clyde A.
CS-Elective 2
11.

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Jurassic Park</title>
<style>
    body {
        font-family: Arial, sans-serif;
        margin: 0;
        padding: 0;
        background-color: #f0f0f0;
    }
    .container {
        max-width: 800px;
        margin: 50px auto;
        padding: 20px;
        background-color: #fff;
        border-radius: 10px;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    h1, h2 {
        text-align: center;
        color: #333;
    }
    p {
        color: #666;
    }
    #imgContainer {
        text-align: center;
    }
    img {
        max-width: 100%;
        height: auto;
    }
    button {
        display: block;
        margin: 0 auto;
        padding: 10px 20px;
        background-color: #007bff;
        color: #fff;
        border: none;
        border-radius: 5px;
        cursor: pointer;
    }
    button:hover {
        background-color: #0056b3;
    }
</style>
</head>
<body>

<div class="container">
    <h1>Welcome to Jurassic Park</h1>
    <div id="imgContainer">
        <img src="https://encrypted-tbn2.gstatic.com/images?q=tbn:ANd9GcSGXk4wQrBRtCQheTNJ6L9LgQ3Ls3l376eMNlZpeTtGTJuJk3Q3" alt="Jurassic Park Poster">
    </div>
    <h2>About Jurassic Park</h2>
    <p>Jurassic Park is a 1993 science fiction adventure film directed by Steven Spielberg and produced by Kathleen Kennedy and Gerald R. Molen. It is the first installment in the Jurassic Park franchise.</p>
    <p>The film is based on the 1990 novel of the same name by Michael Crichton, with a screenplay written by Crichton and David Koepp. It stars Sam Neill, Laura Dern, Jeff Goldblum, Richard Attenborough, Martin Ferrero, BD Wong, Samuel L. Jackson, Wayne Knight, and Joseph Mazzello.</p>
    <button onclick="toggleSynopsis()">Show Synopsis</button>
    <p id="synopsis" style="display: none;">During a preview tour, a theme park suffers a major power breakdown that allows its cloned dinosaur exhibits to run amok.</p>
</div>

<script>
    function toggleSynopsis() {
        var synopsis = document.getElementById("synopsis");
        if (synopsis.style.display === "none") {
            synopsis.style.display = "block";
        } else {
            synopsis.style.display = "none";
        }
    }
</script>

</body>
</html>
```



12.

```python
from sqlalchemy import create_engine, Column, Integer, String
from sqlalchemy.orm import sessionmaker, declarative_base

# Replace 'your_database' with your actual database name
DATABASE_NAME = 'cselect'
DATABASE_URL = f'mysql+mysqlconnector://root:root@127.0.0.1/{DATABASE_NAME}'

# Define the model
Base = declarative_base()

class YourTable(Base):
    __tablename__ = 'clydu'  # Corrected the attribute name
    id = Column(Integer, primary_key=True, autoincrement=True)
    name = Column(String(255))
    age = Column(Integer)
    address = Column(String(255))

# Function to establish a connection to the MySQL database
def connect_to_database():
    engine = create_engine(DATABASE_URL)
    Base.metadata.create_all(engine)  # Create the table if it doesn't exist
    Session = sessionmaker(bind=engine)
    return Session()

# Function to create a new record in the database
def create_record(session, data):
    record = YourTable(**data)
    session.add(record)
    session.commit()
    print("Record inserted successfully.")

# Function to read records from the database
def read_records(session):
    records = session.query(YourTable).all()
    for record in records:
        print(record.__dict__)

# Get user input for data
def get_user_input():
    column1 = input("Enter your Name: ")
    column2 = int(input("Enter your Age: "))
    column3 = input("Enter your address: ")
    return {'name': column1, 'age': column2, 'address': column3}

# Example usage
session = connect_to_database()
if session:
    # Get user input for data
    data_to_insert = get_user_input()

    # Create record with user input data
    create_record(session, data_to_insert)

    print("Records in the database:")
    read_records(session)

    session.close()

```

```
PROBLEMS 1  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\xampp\htdocs\CSElec2> & C:/Users/kentl/AppData/Local/Programs/Python/Python310/python.exe c:/xampp/htdocs/CSElec2/12.py
Enter your Name: Clyde
Enter your Age: 19
Enter your address: San Manuel
Record inserted successfully.
Records in the database:
{'_sa_instance_state': <sqlalchemy.orm.state.InstanceState object at 0x000002A7D3BB9780>, 'address': 'San Manuel', 'id': 1, 'name': 'Clyde', 'age': 19}
PS C:\xampp\htdocs\CSElec2>
```

13.

```python
import requests

class GitHubRepositoryInfoFetcher:
    def __init__(self, owner, repo_name):
        self.owner = owner
        self.repo_name = repo_name
        self.api_url = f'https://api.github.com/repos/{owner}/{repo_name}'

    def fetch_repository_info(self):
        try:
            response = requests.get(self.api_url)
            response.raise_for_status()  # Raises an HTTPError for bad responses

            repo_info = response.json()

            self.display_repository_info(repo_info)
        except requests.exceptions.RequestException as e:
            print(f"Failed to fetch repository information. Error: {e}")

    def display_repository_info(self, repo_info):
        print(f"Repository: {repo_info['name']}")
        print(f"Description: {repo_info['description']}")
        print(f"URL: {repo_info['html_url']}")
        print(f"Language: {repo_info['language']}")
        print(f"Stars: {repo_info['stargazers_count']}")
        print("----")

# Extract owner and repo name from the provided URL
repo_url = "https://github.com/Clyduuu/CSElec2Basig"
owner, repo_name = repo_url.split('/')[-2:]

# Create an instance and fetch/display repository information
github_fetcher = GitHubRepositoryInfoFetcher(owner, repo_name)
github_fetcher.fetch_repository_info()

```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\xampp\htdocs\CSElec2> & C:/Users/kentl/AppData/Local/Programs/Python/Python310/python.exe c:/xampp/htdocs/CSElec2/api.py
Repository: CSElec2Basig
Description: None
URL: https://github.com/Clyduuu/CSElec2Basig
Language: None
Stars: 0
----
PS C:\xampp\htdocs\CSElec2>
```

14.

```python
import unittest

# Function to be tested
def perform_addition(a, b):
    return a + b

# Test cases
class TestOperations(unittest.TestCase):

    # Test case for positive numbers
    def test_add_positive_numbers(self):
        result = perform_addition(3, 4)
        self.assertEqual(result, 7, "Addition of 3 and 4 should be 7")

    # Test case for negative numbers
    def test_add_negative_numbers(self):
        result = perform_addition(-2, -5)
        self.assertEqual(result, -7, "Addition of -2 and -5 should be -7")

    # Test case for one positive and one negative number
    def test_add_mixed_numbers(self):
        result = perform_addition(8, -3)
        self.assertEqual(result, 5, "Addition of 8 and -3 should be 5")

# Run the tests
if __name__ == '__main__':
    unittest.main()

```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\xampp\htdocs\CSElec2> & C:/Users/kentl/AppData/Local/Programs/Python/Python310/python.exe c:/xampp/htdocs/CSElec2/14.py
...
----------------------------------------------------------------------
Ran 3 tests in 0.001s

OK
PS C:\xampp\htdocs\CSElec2>
```