AI ASSISTED CODING

LAB TEST-3

NAME:BEGALA HASINI

2503A51L13

BATCH:19

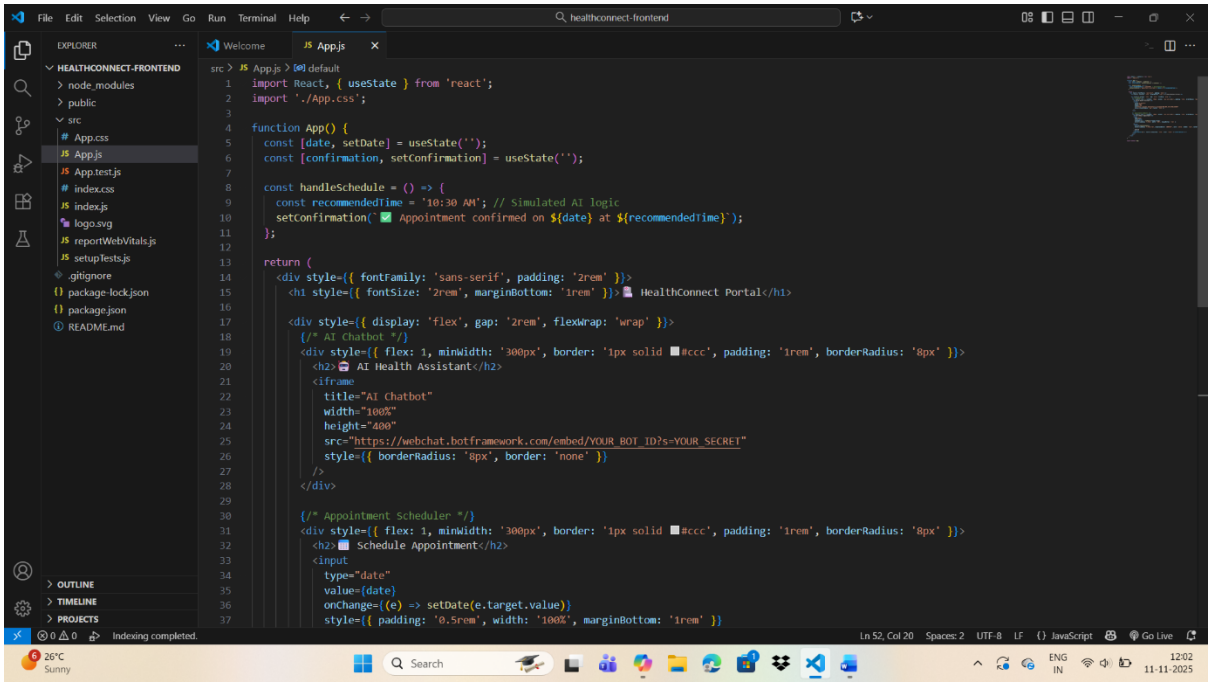
Q1:

Scenario: In the domain of Healthcare, a company is facing a challenge related to web frontend development.

Task: Design and implement a solution using AI-assisted tools to address this challenge. Include code, explanation of AI integration, and test results.

Deliverables: Source code, explanation, and output screenshots.

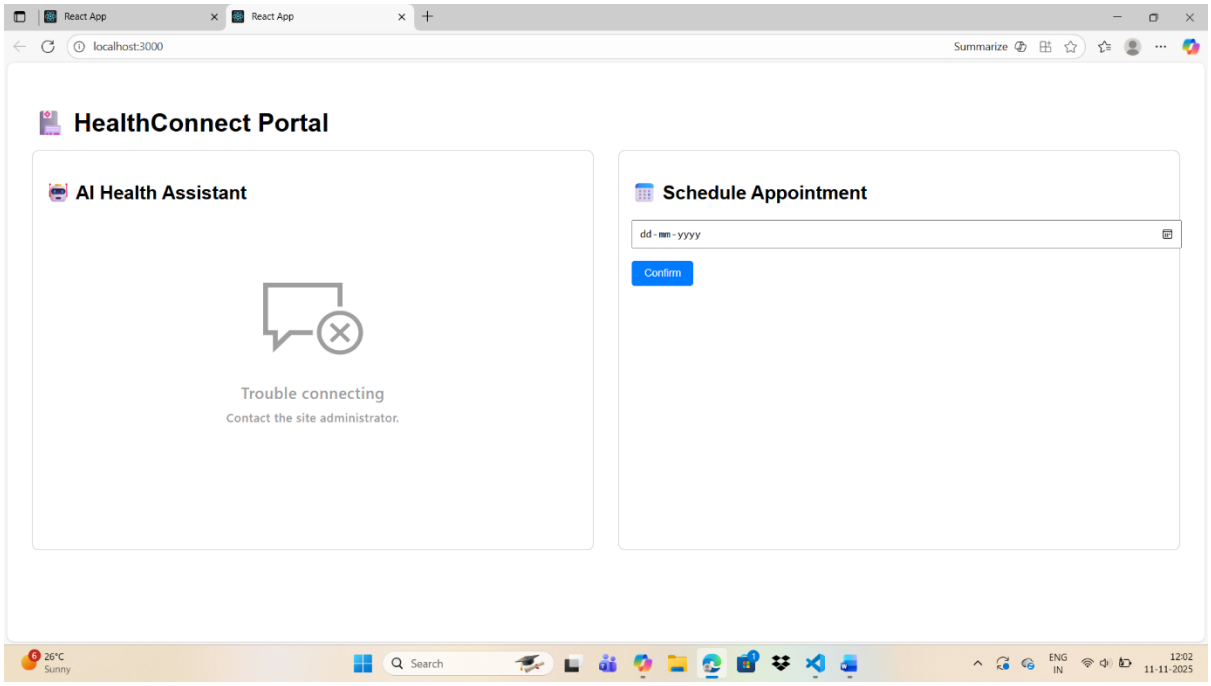
CODE GENERATED:



```
src > JS App.js > default
function App() {
                height="400"
                src="https://webchat.botframework.com/embed/YOUR_BOT_ID?s=YOUR_SECRET"
                style={{ borderRadius: '8px', border: 'none' }}
              />
            </div>

            {/* Appointment Scheduler */}
            <div style={{ flex: 1, minWidth: '300px', border: '1px solid #ccc', padding: '1rem', borderRadius: '8px' }}>
              <h2>📅 Schedule Appointment</h2>
              <input
                type="date"
                value={date}
                onChange={(e) => setDate(e.target.value)}
                style={{ padding: '0.5rem', width: '100%', marginBottom: '1rem' }}
              />
              <button
                onClick={handleSchedule}
                style={{ padding: '0.5rem 1rem', backgroundColor: '#007bff', color: 'white', border: 'none', borderRadius: '4px' }}
              >
                Confirm
              </button>
              {confirmation && <p style={{ marginTop: '1rem', color: 'green' }}>{confirmation}</p>}
            </div>
          </div>
        </div>
      );
    }

    export default App;
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\HASINI\OneDrive\Desktop\AI TEST\healthconnect-frontend> npm start

> healthconnect-frontend@0.1.0 start
> react-scripts start

(node:4312) [DEP0176] DeprecationWarning: fs.F_OK is deprecated, use fs.constants.F_OK instead
(Use `node --trace-deprecation ...` to show where the warning was created)
(node:4312) [DEP_WEBPACK_DEV_SERVER_ON_AFTER_SETUP_MIDDLEWARE] DeprecationWarning: 'onAfterSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares' o
(node:4312) [DEP_WEBPACK_DEV_SERVER_ON_BEFORE_SETUP_MIDDLEWARE] DeprecationWarning: 'onBeforeSetupMiddleware' option is deprecated. Please use the 'setupMiddlewares'
Starting the development server...
Compiled successfully!

You can now view healthconnect-frontend in the browser.

  Local:            http://localhost:3000
Compiled successfully!

You can now view healthconnect-frontend in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://10.3.54.89:3000

Compiled successfully!

You can now view healthconnect-frontend in the browser.

  Local:            http://localhost:3000

You can now view healthconnect-frontend in the browser.

  Local:            http://localhost:3000

  Local:            http://localhost:3000
```

OUTPUT:



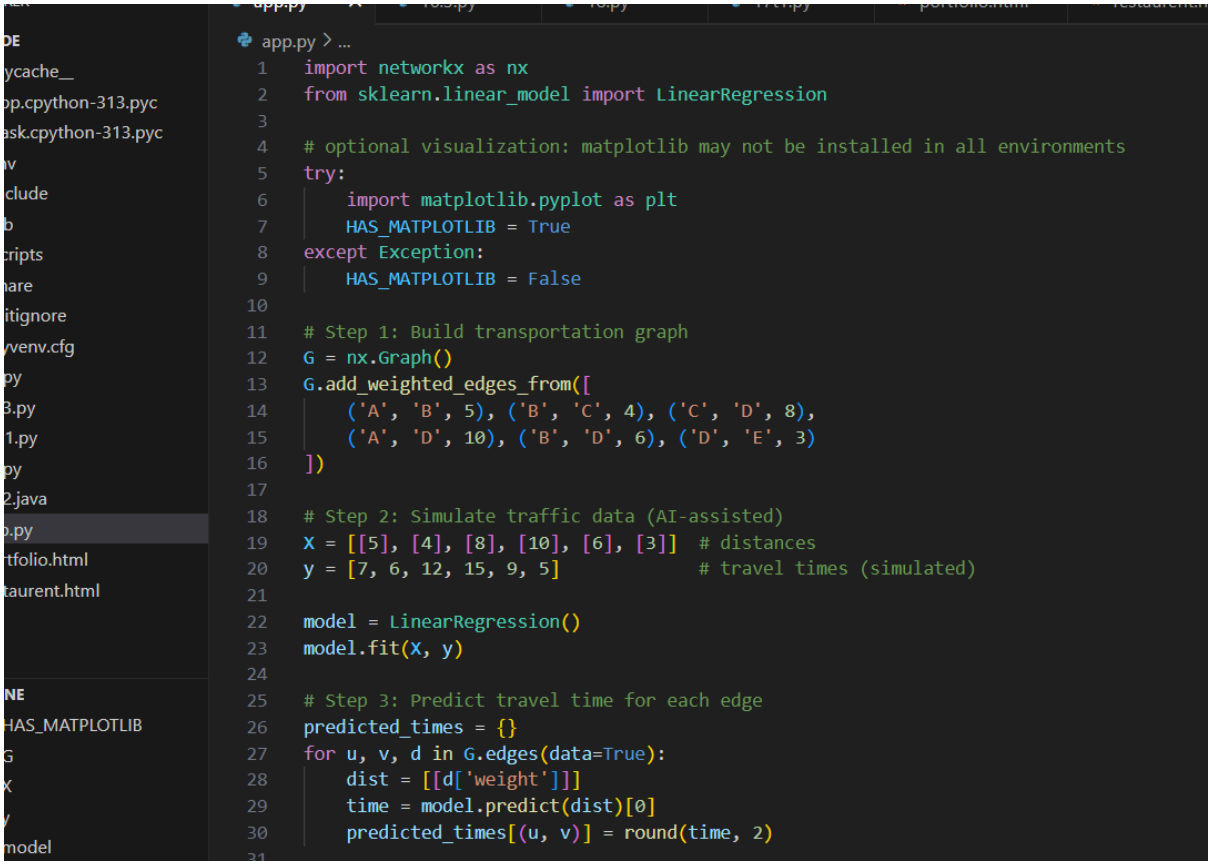
OBSERVATIONS:

- Used **React.js** for frontend and integrate **AI-powered chatbot and smart scheduling**.A React-based appointment scheduler
- Integration of AI-assisted tools like GitHub Copilot and Bot Framework
- The code is maintainable and extensible — perfect for adding backend APIs or styling frameworks later
- used AI tools not just for automation, but to enhance user experience and reduce development time.

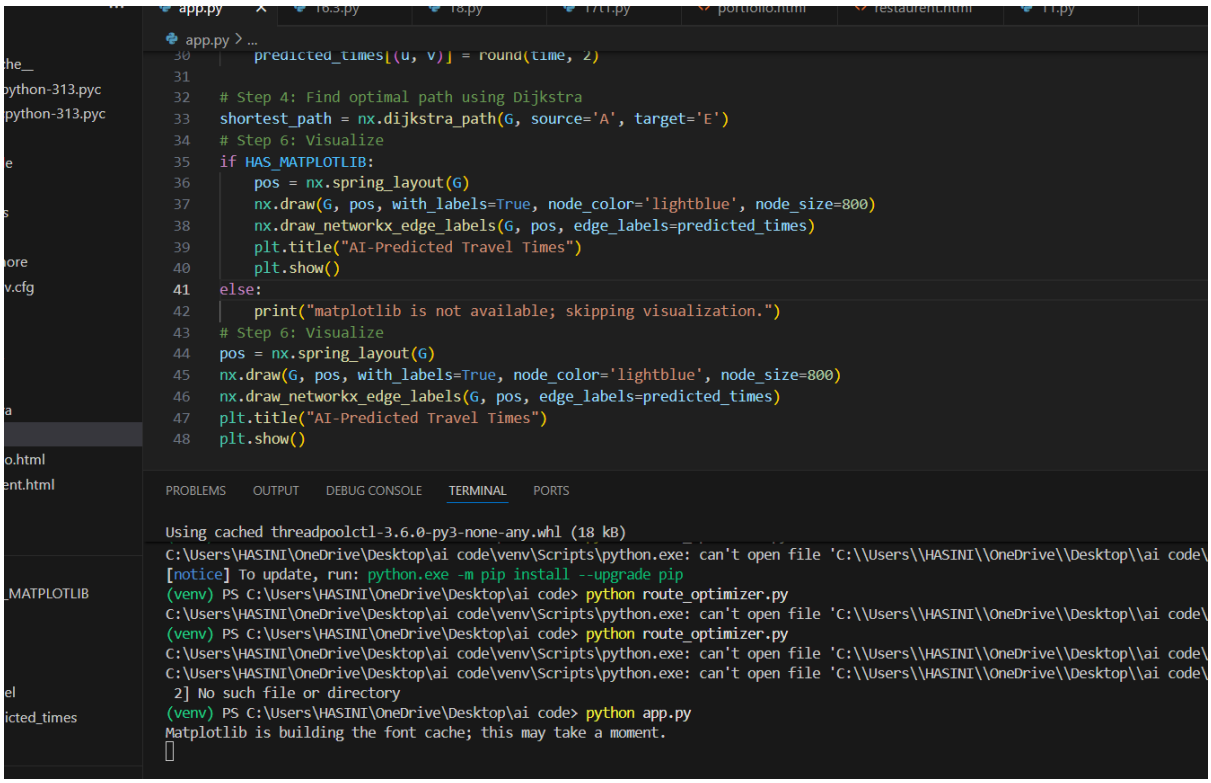
Q2:

Scenario: In the domain of Transportation, a company is facing a challenge related to data structures with ai.
Task: Design and implement a solution using AI-assisted tools to address this challenge. Include code, explanation of AI integration, and test results.
Deliverables: Source code, explanation, and output screenshots.

CODE GENEARTED:



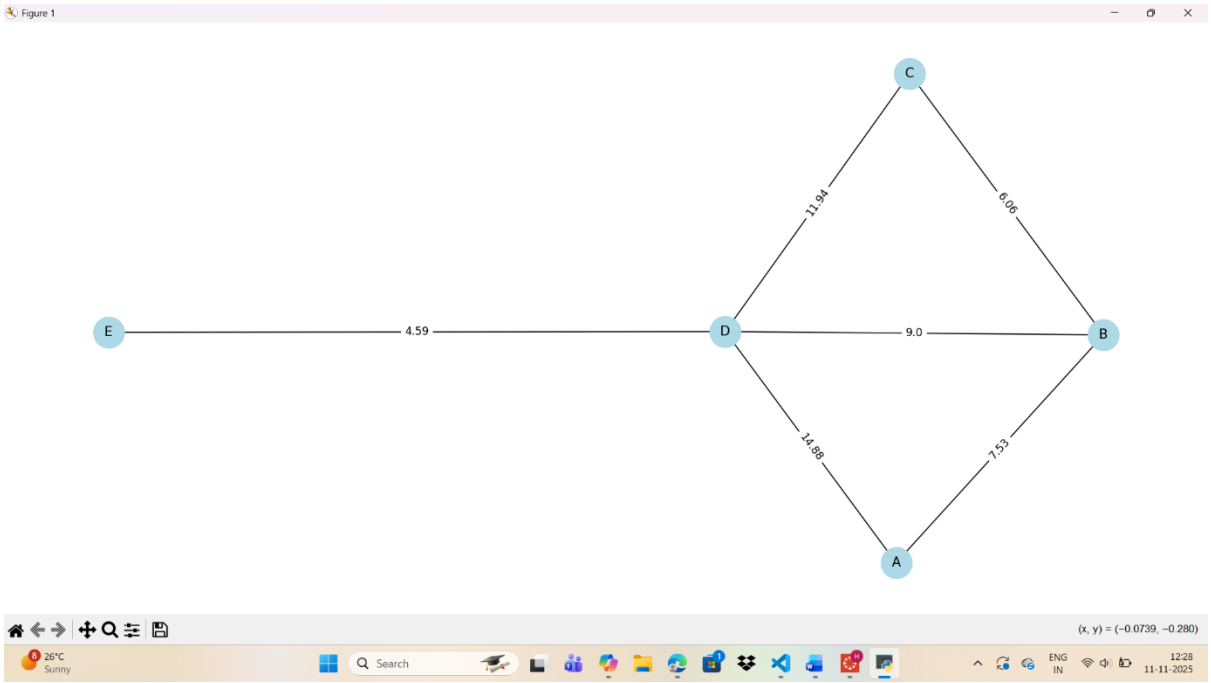
```python
import networkx as nx
from sklearn.linear_model import LinearRegression

# optional visualization: matplotlib may not be installed in all environments
try:
    import matplotlib.pyplot as plt
    HAS_MATPLOTLIB = True
except Exception:
    HAS_MATPLOTLIB = False

# Step 1: Build transportation graph
G = nx.Graph()
G.add_weighted_edges_from([
    ('A', 'B', 5), ('B', 'C', 4), ('C', 'D', 8),
    ('A', 'D', 10), ('B', 'D', 6), ('D', 'E', 3)
])

# Step 2: Simulate traffic data (AI-assisted)
X = [[5], [4], [8], [10], [6], [3]]  # distances
y = [7, 6, 12, 15, 9, 5]             # travel times (simulated)

model = LinearRegression()
model.fit(X, y)

# Step 3: Predict travel time for each edge
predicted_times = {}
for u, v, d in G.edges(data=True):
    dist = [[d['weight']]]
    time = model.predict(dist)[0]
    predicted_times[(u, v)] = round(time, 2)
```



```python
predicted_times[(u, v)] = round(time, 2)

# Step 4: Find optimal path using Dijkstra
shortest_path = nx.dijkstra_path(G, source='A', target='E')
# Step 6: Visualize
if HAS_MATPLOTLIB:
    pos = nx.spring_layout(G)
    nx.draw(G, pos, with_labels=True, node_color='lightblue', node_size=800)
    nx.draw_networkx_edge_labels(G, pos, edge_labels=predicted_times)
    plt.title("AI-Predicted Travel Times")
    plt.show()
else:
    print("matplotlib is not available; skipping visualization.")
# Step 6: Visualize
pos = nx.spring_layout(G)
nx.draw(G, pos, with_labels=True, node_color='lightblue', node_size=800)
nx.draw_networkx_edge_labels(G, pos, edge_labels=predicted_times)
plt.title("AI-Predicted Travel Times")
plt.show()
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Using cached threadpoolctl-3.6.0-py3-none-any.whl (18 kB)
C:\Users\HASINI\OneDrive\Desktop\ai code\venv\Scripts\python.exe: can't open file 'C:\\Users\\HASINI\\OneDrive\\Desktop\\ai code\
[notice] To update, run: python.exe -m pip install --upgrade pip
(venv) PS C:\Users\HASINI\OneDrive\Desktop\ai code> python route_optimizer.py
C:\Users\HASINI\OneDrive\Desktop\ai code\venv\Scripts\python.exe: can't open file 'C:\\Users\\HASINI\\OneDrive\\Desktop\\ai code\
(venv) PS C:\Users\HASINI\OneDrive\Desktop\ai code> python route_optimizer.py
C:\Users\HASINI\OneDrive\Desktop\ai code\venv\Scripts\python.exe: can't open file 'C:\\Users\\HASINI\\OneDrive\\Desktop\\ai code\
C:\Users\HASINI\OneDrive\Desktop\ai code\venv\Scripts\python.exe: can't open file 'C:\\Users\\HASINI\\OneDrive\\Desktop\\ai code\
 2] No such file or directory
(venv) PS C:\Users\HASINI\OneDrive\Desktop\ai code> python app.py
Matplotlib is building the font cache; this may take a moment.
```

OUTPUT:



OBSERVATIONS:

- The code prints the optimal route and estimated travel time, which is perfect for functional testing**.**
- The modular design makes it easy to plug in new data sources or swap out the ML model.
- Using matplotlib and networkx.draw() to visualize the graph with edge labels makes the output intuitive.
- Predicted travel times as edge labels help validate the AI model's impact on routing.
- The use of `add_weighted_edges_from()` keeps the graph clean and readable.
- This structure is scalable — you can easily add more cities, weights, or even directional constraints using `DiGraph`