

LAB 1

Inayat arshad

```
//code
```

```
#include <iostream>
```

```
#include <fstream>
```

```
#include <vector>
```

```
#include <string>
```

```
#include <filesystem>
```

```
#ifdef _WIN32
```

```
#include <windows.h>
```

```
#endif
```

```
using namespace std;
```

```
int main() {
```

```
    int rows, cols;
```

```
    cout << "Enter number of rows: ";
```

```
    cin >> rows;
```

```
    cout << "Enter number of columns: ";
```

```
    cin >> cols;
```

```
    // Consume leftover newline
```

```
    cin.ignore();
```

```
    // 2D vector to store table data
```

```

vector<vector<string>> table(rows, vector<string>(cols));

// Input cell-by-cell
for (int i = 0; i < rows; ++i) {
    for (int j = 0; j < cols; ++j) {
        cout << "Enter data for cell (" << i + 1 << ", " << j + 1 << "): ";
        getline(cin, table[i][j]);
    }
}

// Determine Desktop path and create folder
string folderPath;

#ifdef _WIN32
    char desktopPath[MAX_PATH];
    SHGetFolderPathA(NULL, CSIDL_DESKTOP, NULL, 0, desktopPath);
    folderPath = string(desktopPath) + "\\HTMLTableOutput";
    filesystem::create_directory(folderPath);
    folderPath += "\\table.html";
#else
    folderPath = getenv("HOME") + string("/Desktop/HTMLTableOutput/table.html");
    filesystem::create_directories(filesystem::path(folderPath).parent_path());
#endif

// Generate HTML
ofstream htmlFile(folderPath);

if (!htmlFile) {

```

```

    cerr << "Error creating HTML file.\n";

    return 1;
}

htmlFile << "<!DOCTYPE html>\n<html>\n<head>\n<title>Generated Table</title>\n"

    << "<style>table, th, td { border: 1px solid black; border-collapse: collapse; padding:
8px; }</style>\n"

    << "</head>\n<body>\n<h2>Generated Table</h2>\n<table>\n";

for (const auto& row : table) {
    htmlFile << "<tr>";

    for (const auto& cell : row) {
        htmlFile << "<td>" << cell << "</td>";
    }

    htmlFile << "</tr>\n";
}

htmlFile << "</table>\n</body>\n</html>";

htmlFile.close();

cout << "HTML table has been generated and saved at:\n" << folderPath << endl;

return 0;
}

```