Building Material Scoring System

Portfolio Project # 1

Muhammad Bilal

Building Material Scoring System

This Python script provides a system for calculating scores based on different types of building materials used in a construction project. The script utilizes the Tkinter library for creating a simple graphical user interface (GUI) to upload a file containing building material data. After uploading the file, the script calculates scores for glass, recycled, stone, and wood materials, and then writes the results to a text file.

Prerequisites

- Python 3.x
- Tkinter library (usually included in Python standard library)

Usage

1. **Upload File**: Click the "Upload File" button to select a file containing building material data. The file should have a specific format where building materials are represented as codes (e.g., G10 for glass with a score of 10).
2. **View Results**: Once the file is uploaded, the script calculates scores for each type of building material and the total score. The results are written to a text file named **scoring-results.txt**, which is saved in the **datafiles** directory.
3. **Result File**: The **scoring-results.txt** file contains a breakdown of scores for each material type as well as the total score.

File Format

- The input file should be a text file where each line represents a row in the building.
- Building materials are represented by codes:
  - **G**: Glass
  - **R**: Recycled
  - **S**: Stone
  - **W**: Wood
- The code is followed by a numerical value representing the score for that material.
- Materials in each row are separated by the pipe character (**|**).

Example of valid input:

Copy code

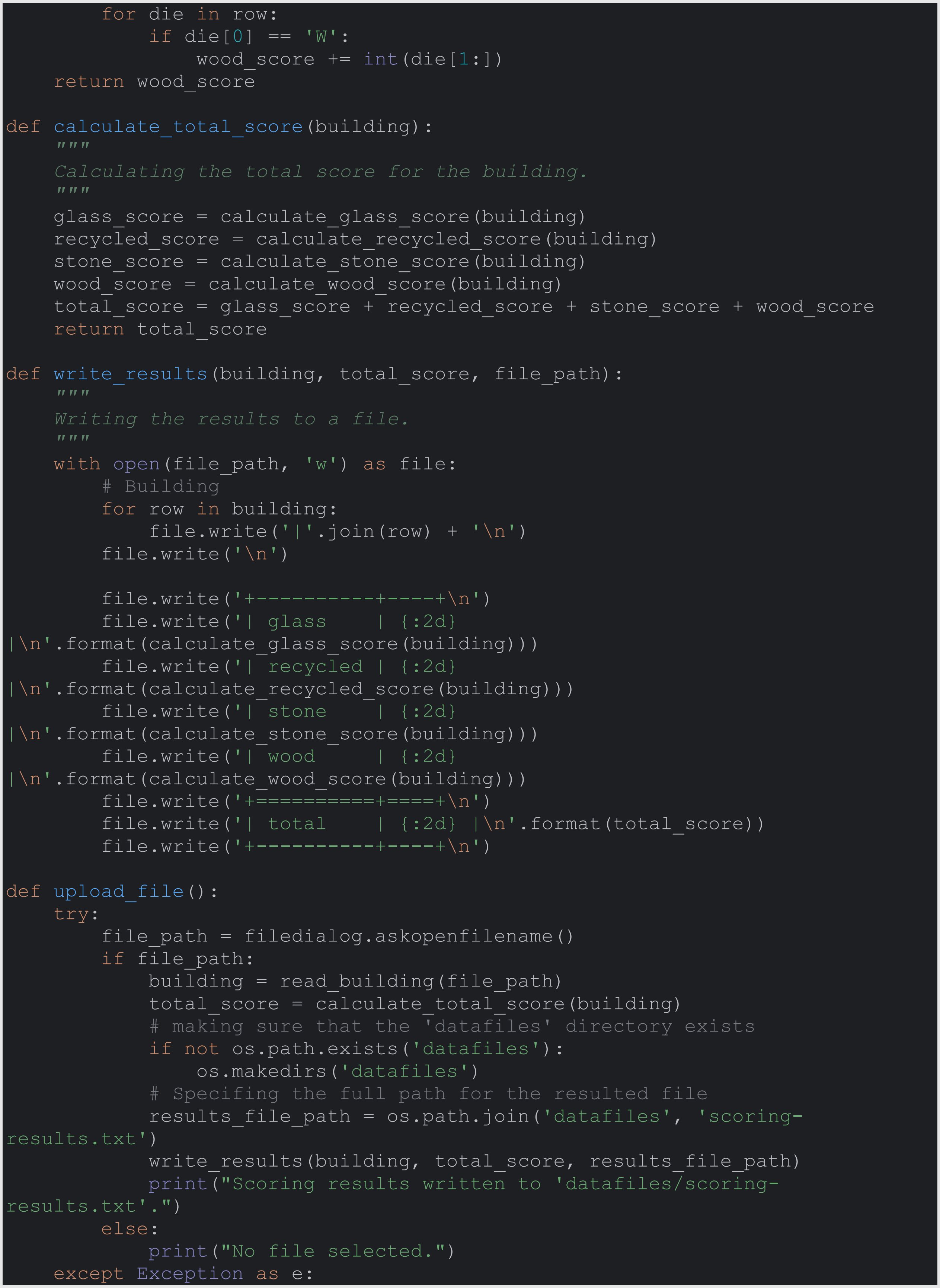
```
G10|R5|S8|W3 R12|G6|S4|W9 S7|W5|G3|R10
```

Note

- This script assumes that the input file follows the specified format. Any deviation from the format may result in incorrect results or errors.

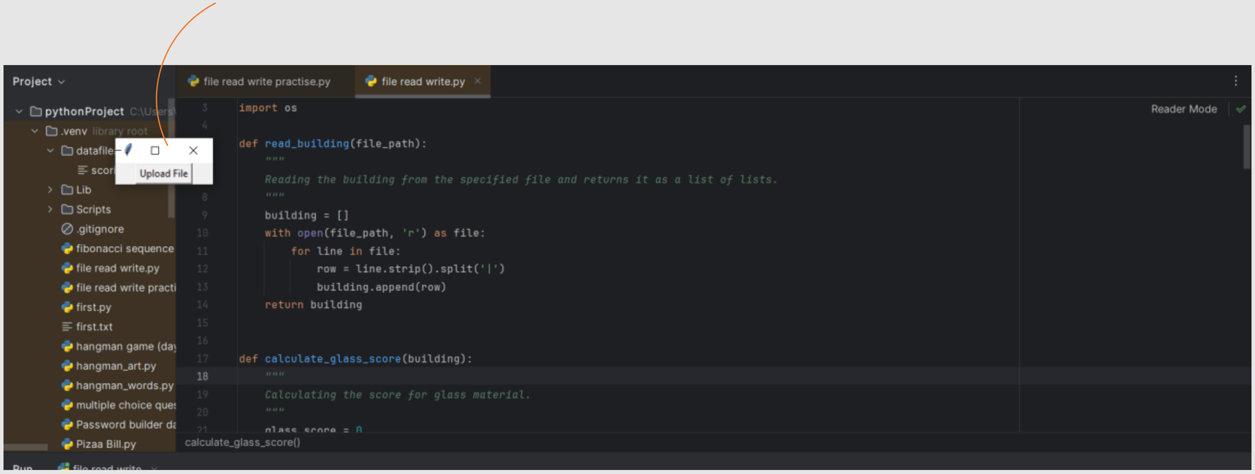
Code:

```
import tkinter as tk
from tkinter import filedialog
import os

def read_building(file_path):
    """
    Reading the building from the specified file and returns it as a list of
lists.
    """
    building = []
    with open(file_path, 'r') as file:
        for line in file:
            row = line.strip().split('|')
            building.append(row)
    return building


def calculate_glass_score(building):
    """
    Calculating the score for glass material.
    """
    glass_score = 0
    for row in building:
        for die in row:
            if die[0] == 'G':
                glass_score += int(die[1:])
    return glass_score

def calculate_recycled_score(building):
    """
    Calculating the score for recycled material.
    """
    recycled_score = 0
    for row in building:
        for die in row:
            if die[0] == 'R':
                recycled_score += int(die[1:])
    return recycled_score

def calculate_stone_score(building):
    """
    Calculating the score for stone material.
    """
    stone_score = 0
    for row in building:
        for die in row:
            if die[0] == 'S':
                stone_score += int(die[1:])
    return stone_score

def calculate_wood_score(building):
    """
    Calculating the score for wood material.
    """
    wood_score = 0
    for row in building:
```



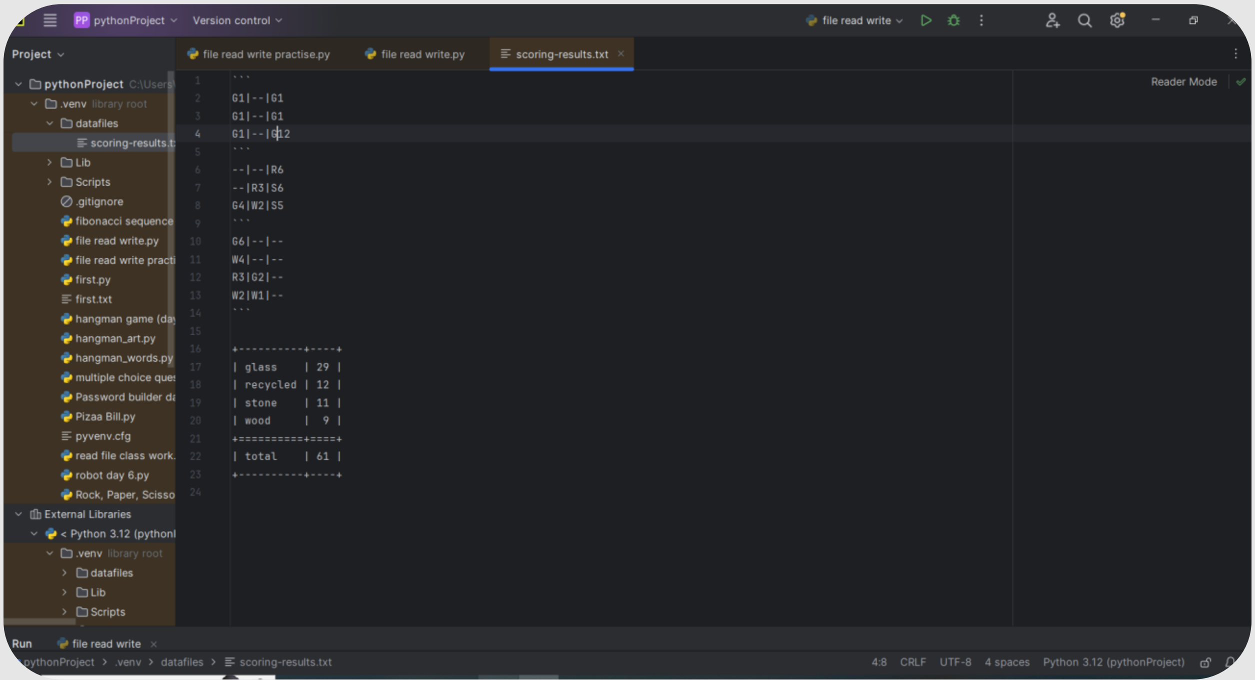
```
        for die in row:
            if die[0] == 'W':
                wood_score += int(die[1:])
    return wood_score

def calculate_total_score(building):
    """
    Calculating the total score for the building.
    """
    glass_score = calculate_glass_score(building)
    recycled_score = calculate_recycled_score(building)
    stone_score = calculate_stone_score(building)
    wood_score = calculate_wood_score(building)
    total_score = glass_score + recycled_score + stone_score + wood_score
    return total_score

def write_results(building, total_score, file_path):
    """
    Writing the results to a file.
    """
    with open(file_path, 'w') as file:
        # Building
        for row in building:
            file.write('|'.join(row) + '\n')
        file.write('\n')

        file.write('+----------+-----+\n')
        file.write('| glass    | {:2d} 
|\n'.format(calculate_glass_score(building)))
        file.write('| recycled | {:2d} 
|\n'.format(calculate_recycled_score(building)))
        file.write('| stone    | {:2d} 
|\n'.format(calculate_stone_score(building)))
        file.write('| wood     | {:2d} 
|\n'.format(calculate_wood_score(building)))
        file.write('+==========+=====+\n')
        file.write('| total    | {:2d} |\n'.format(total_score))
        file.write('+----------+-----+\n')

def upload_file():
    try:
        file_path = filedialog.askopenfilename()
        if file_path:
            building = read_building(file_path)
            total_score = calculate_total_score(building)
            # making sure that the 'datafiles' directory exists
            if not os.path.exists('datafiles'):
                os.makedirs('datafiles')
            # Specifing the full path for the resulted file
            results_file_path = os.path.join('datafiles', 'scoring-
results.txt')
            write_results(building, total_score, results_file_path)
            print("Scoring results written to 'datafiles/scoring-
results.txt'.")
        else:
            print("No file selected.")
    except Exception as e:
        print("An error occurred:", e)
    finally:
        root.destroy()  # Close the Tkinter application window after uploading the file

root = tk.Tk()
root.title("File Upload")

upload_button = tk.Button(root, text="Upload File", command=upload_file)
upload_button.pack()

root.mainloop()
```

File uploading Button:



```
import os

def read_building(file_path):
    """
    Reading the building from the specified file and returns it as a list of lists.
    """
    building = []
    with open(file_path, 'r') as file:
        for line in file:
            row = line.strip().split('|')
            building.append(row)
    return building


def calculate_glass_score(building):
    """
    Calculating the score for glass material.
    """
    glass_score = 0
```

Result:



~~~
```
G1|--|G1
G1|--|G1
G1|--|G12
```
--|--|R6
--|R3|S6
G4|W2|S5
```
G6|--|--
W4|--|--
R3|G2|--
W2|W1|--
```


+----------+----+
| glass    | 29 |
| recycled | 12 |
| stone    | 11 |
| wood     |  9 |
+==========+====+
| total    | 61 |
+----------+----+
~~~