

Room Allocation Web Application Documentation

Logic Explanation:-

app.py:-

Flask Setup

- **import Flask, render_template, request, jsonify**
 - Flask classes for the making of the web application, templates, request and response using json.
- **app = Flask(__name__)**
 - initializes the Flask application.

Index Route

- **@app.route('/')**
 - Defines the route for home page.
- **def index(): return render_template('index.html')**
 - this renders the index.html template, which contains file upload form.

Upload Route

- **@app.route('/upload', methods=['POST'])**
 - Defines the route for handling file uploads via POST method.
- **def upload_files():**
 - The upload_files function handles the uploaded CSV files.
 - group_file = request.files['group_file'] and hostel_file = request.files['hostel_file'] retrieves the uploaded files from the request.
 - Uses Pandas to read the CSV files: group_df = pd.read_csv(group_file) and hostel_df = pd.read_csv(hostel_file).
 - Calls the allocate_rooms function to process the data.
 - Displays the allocation results on the webpage.

Room Allocation Logic

allocate_rooms(group_df, hostel_df)

- **def allocate_rooms(group_df, hostel_df):**
 - allocations = [] initializes an empty list to store the allocation results.

Processing Groups

- **for _, group in group_df.iterrows():**
 - Iterates over each row in the group dataframe.
 - Extracts the group ID, number of members, and gender from the current row.

Handling Mixed Gender Groups

- Checks if the gender column contains both boys and girls: if **isinstance(gender, str)** and **'&'** in gender:
 - Splits the group into separate boy and girl subgroups if mixed.
 - Creates separate dictionaries for boys and girls with the same group ID but different member counts and genders.

Allocating Rooms

- for **sub_group in group_list**:
 - Iterates over each subgroup (either a single gender group or a split mixed gender group).
 - Extracts the subgroup ID, number of members, and gender.
 - for **_, room in hostel_df.iterrows()**:
 - Iterates over each row in the hostel dataframe to find a suitable room.
 - Checks if the room can accommodate the subgroup: if **room['Gender'] == sub_gender and room['Capacity'] >= sub_members**:
 - Adds the allocation details to the allocations list.
 - Updates the room capacity in the dataframe to reflect the allocated members.