**House Remodeling Supply Tracker**

**Objective:**

You will build a Python web application using Flask and SQLAlchemy to track the supplies and costs needed for remodeling various rooms in a house. The application will allow users to enter detailed information about each room, including the type of remodeling (tiling, flooring, painting, etc.) and the materials required. The app will calculate total remodeling costs, display detailed breakdowns, and visualize data using Seaborn bar graphs.

This project will require:

* SQLAlchemy models with foreign key relationships.
* Instantiable classes and complex data handling using dictionaries, tuples, and lists.
* Unit tests using PyTest to validate application functionality.
* A web interface to add, edit, and display room and supply data.
* A Seaborn bar graph to visually represent the data.

**Project Requirements:**

**1. SQLAlchemy Models:**

You will create two main models: **Room** and **Supply**. Each room will have a list of associated supplies and costs.

**Room Model:**

* id: Primary key (integer).
* name: The name of the room (string).
* surface\_area: Total surface area of the room in square feet (float).
* flooring\_type: Type of flooring (e.g., "Hardwood", "Tile") (string).
* flooring\_cost\_per\_sqft: Cost of the flooring per square foot (float).
* **New Fields**:
  + is\_tiling\_needed: Boolean indicating whether tiling is needed (boolean).
  + tile\_type: Type of tile (e.g., "Ceramic", "Porcelain") (string).
  + tile\_cost\_per\_sqft: Cost of tiling per square foot (float).
  + tiling\_area: Area of the room that requires tiling (float).
  + total\_tile\_cost: Calculated field for the total cost of tiling (tiling\_area \* tile\_cost\_per\_sqft).
  + total\_flooring\_cost: Total cost for the flooring, calculated as surface\_area \* flooring\_cost\_per\_sqft.
  + total\_remodel\_cost: Calculated field for the total remodeling cost (sum of flooring, tiling, and supply costs).

**Supply Model:**

* id: Primary key (integer).
* room\_id: Foreign key linking to the Room model.
* name: Name of the supply (string).
* quantity: Quantity of the supply needed (integer).
* cost\_per\_item: Cost of each supply item (float).
* total\_supply\_cost: Calculated field for the total cost of the supply (quantity \* cost\_per\_item).

**Foreign Key Relationship:**

* room\_id in Supply will be a foreign key that references id in the Room model.

**2. Web Application Pages:**

**Home Page:**

* Lists all the rooms that are being remodeled.
* Displays total remodeling costs for each room.

**Add Room Page:**

* Form to add a new room, capturing:
  + Room name.
  + Surface area in square feet.
  + Flooring type and cost per square foot.
  + Checkbox for tiling requirement (if applicable).
  + Tile type, cost per square foot, and tiling area (if applicable).
  + Button to submit room details.

**Edit Room Page:**

* Form pre-filled with existing room data to allow updating information.

**Room Details Page:**

* Displays the details for a specific room, including:
  + Flooring type and cost.
  + Tiling information (if tiling is required).
  + Total costs for flooring, tiling, and supplies.
  + A list of all associated supplies and their costs.
  + A total remodeling cost for the room, which is:
    - total\_remodel\_cost = total\_flooring\_cost + total\_tile\_cost + sum(supply.total\_supply\_cost for supply in room.supplies)

**Add Supply Page:**

* Form to add supplies to a specific room:
  + Supply name.
  + Quantity.
  + Cost per item.
  + Button to submit supply details.

**Supply Details Page:**

* Displays details of a specific supply, including total cost.

**3. Advanced Calculations:**

**Total Room Remodeling Cost:**

* + total\_remodel\_cost = total\_flooring\_cost + total\_tile\_cost + sum(supply.total\_supply\_cost for supply in room.supplies)

This calculation includes:

* Flooring cost (surface\_area \* flooring\_cost\_per\_sqft).
* Tiling cost (if applicable: tiling\_area \* tile\_cost\_per\_sqft).
* Supply costs (quantity \* cost\_per\_item for each supply).

**Tile Needed Calculation:**

* total\_tile\_cost = tiling\_area \* tile\_cost\_per\_sqft

**4. Seaborn Bar Graph Visualization:**

You will create a Seaborn bar graph on a separate page that compares the total tiling costs or total remodeling costs across all rooms.

* **X-axis**: Room names.
* **Y-axis**: Total tiling cost (or total remodeling cost).
* This graph will allow users to quickly visualize and compare costs between rooms.

**5. Unit Testing with PyTest:**

You will create unit tests to ensure the correctness of your calculations and functionality.

* You are developing your code. You need to be able to estimate what 80% coverage is based on what you build. It is your job as a developer to develop that deeper level of understanding.