

BIL211 2021 Spring

Lab Assignment #3

Deadline: 22.02.2021 12:30 GMT +3

❖ Implement the following classes:

Class Name: Estate (implements Comparable)

- address : String
 - floor : int
 - facade : {"North", "East", "South", "West"}
 - bedrooms : int
 - aream2 : int
 - quality : [1-5] (Higher is better)
- ❖ getPrice() : returns int.
- This is to be overridden in inherited classes.
- ❖ compareTo (Estate p, String s) : returns {-1, 0, 1}.
- If String s is "Price": If the estate's calculated price is higher than p's returns 1, if the opposite case returns -1, and if the prices are equal it returns 0.
 - If String s is "Quality": If the estate's quality is better/higher than p's returns 1, if the opposite case returns -1, and if the qualities are equal it returns 0.
 - If String s is "Bedrooms": If the number of bedrooms in the estate is higher than p's returns 1, if the opposite case returns -1, and if the number of bedrooms are equal it returns 0.
 - If String s is "Area": If the estate's area in m² is bigger than p's returns 1, if the opposite case returns -1, and if the areas are equal it returns 0.
- ❖ toString() : returns String.
- toString function returns estate info as String. Each estate can have different attributes.
- ❖ copy() : returns a deep copy of the caller Estate object.

Class Name: Apartment (extends Estate)

- address : String
 - Address is used to identify the building, doesn't have an effect on price.
- floor : int
 - The floor the apartment belongs in, can be negative.
 - Adds 30,000 to the price of the apartment for every floor that is higher than ground level. (floor 0 adds 0, floor 2 adds 60,000, -1 subtracts 30,000)
- facade : {"North", "East", "South", "West"}
 - "South": adds 20,000 to the price.
 - "North": subtracts 20,000 from the price.
 - "West": adds 5,000 to the price.
 - "East": subtracts 5,000 from the price.
- bedrooms : int
 - Adds 35,000 to the price for every room.
- aream2 : int
 - Adds 1,000 to the price for every 5 m².
- quality : [1-5] (Higher is better)
 - Each quality point adds 60,000 to the price.
- ❖ getPrice() : returns int.
 - This is to be overridden in inherited classes.
- ❖ compareTo (Estate p) : returns {-1, 0, 1}.
 - If String s is "Price": If the apartment's calculated price is higher than p's returns 1, if the opposite case returns -1, and if the prices are equal it returns 0.
 - If String s is "Quality": If the apartment's quality is better/higher than p's returns 1, if the opposite case returns -1, and if the qualities are equal it returns 0.
 - If String s is "Bedrooms": If the number of bedrooms in the apartment is higher than p's returns 1, if the opposite case returns -1, and if the number of bedrooms are equal it returns 0.
 - If String s is "Area": If the apartment's area in m² is higher than p's returns 1, if the opposite case returns -1, and if the areas are equal it returns 0.
- ❖ toString() : returns String.
 - toString function returns estate info as String. Each estate can have different attributes.
 - Output example:
Address: ABC St, 06345, Ank
Floor: 3
Facade: East
Number of bedrooms: 3
Area in m2: 123
Quality: 3
Price: 500000
- ❖ copy() : returns a deep copy of the caller Apartment object.

Class Name: House (extends Estate)

- address : String
 - Address is used to identify the building, doesn't have an effect on price.
- floor : int
 - *floor* represents how many floors the house have, must be nonnegative.
 - Each additional floor adds 100,000 to the price.
- facade : {"North", "East", "South", "West"}
 - Regardless of the facade, the price of any house is increased by 150,000.
- bedrooms : int
 - Adds 65,000 to the price for every room.
- aream2 : int
 - Adds 3,000 to the price for every 5 m².
- quality : [1-5] (Higher is better)
 - Each quality point adds 110,000 to the price for every unit.
- ❖ getPrice() : returns int.
 - This is to be overridden in inherited classes.
- ❖ compareTo (Estate p, "Price") : returns {-1, 0, 1}.
 - If String s is "Price": If the house's calculated price is higher than p's returns 1, if the opposite case returns -1, and if the prices are equal it returns 0.
 - If String s is "Quality": If the house's quality is better/higher than p's returns 1, if the opposite case returns -1, and if the qualities are equal it returns 0.
 - If String s is "Bedrooms": If the number of bedrooms in the house is higher than p's returns 1, if the opposite case returns -1, and if the number of bedrooms are equal it returns 0.
 - If String s is "Area": If the house's area in m² is higher than p's returns 1, if the opposite case returns -1, and if the areas are equal it returns 0.
- ❖ toString() : returns String.
 - toString function returns estate info as String. Each estate can have different attributes.
 - Output example:
Address: ABC St, 06345, Ank
Number of floors: 3
Facade: East
Number of bedrooms: 3
Area in m2: 123
Quality: 3
Price: 500000
- ❖ copy() : returns a deep copy of the caller House object.

- You **must** use getter and setter methods, a.k.a. your variables should be private.
- You **cannot** change given Java classes.
- Github: <https://classroom.github.com/a/QM6GTU3X>
- Your commits **should not include** RealEstate.java. In case of an accident, [revert the said commit](#).
- You don't need to validate the correctness of the given inputs.