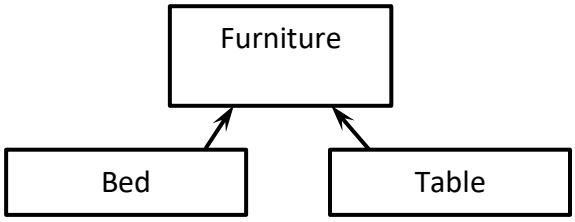
# Programming Assignment:

# Furniture Inheritance



In this assignment, you will be creating an inheritance hierarchy of classes used to handle the different types of furniture items in a store. The class hierarchy is shown in the following diagram:



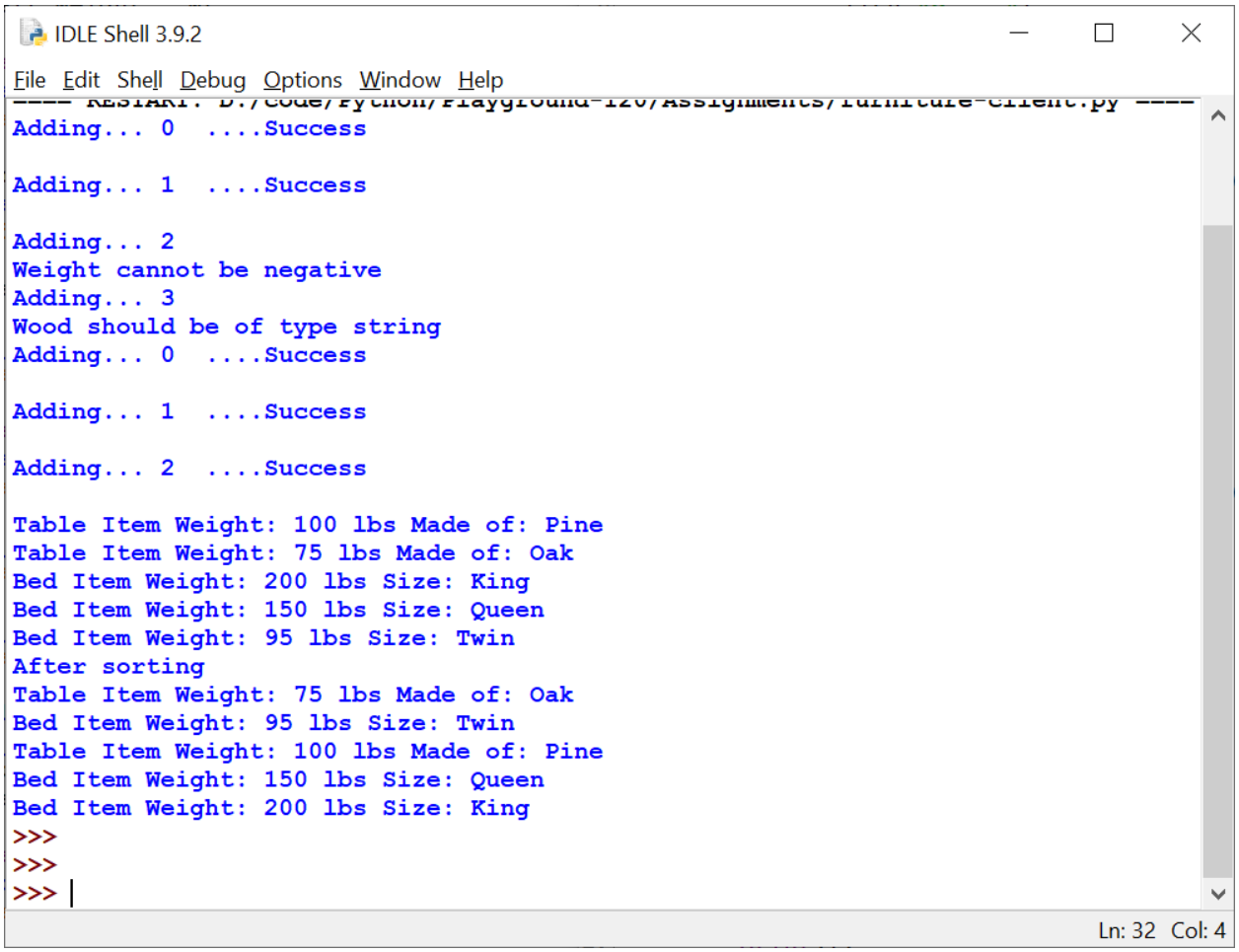

Specifically,

1) Add a base class called **Furniture**:
    a. Add a private attribute called **__weight** which represents the weight of the furniture in lbs.
    b. Add a constructor that takes one argument, weight in lbs. The constructor should set the private attribute called **__weight**. It should throw a ValueError exception saying "Weight must be positive" if the weight argument is less than or equal to 0.
    c. Add a public property called **weight** backed by **__weight** private attribute. Setter of this property should make sure new value is positive, otherwise should throw a ValueError exception saying "Weight must be positive" and NOT update the **__weight** attribute.
    d. Add a **__str__** method that returns a string of the form "Item Weight: *weight* "
    e. Make the Furniture class comparable. The comparison should be based on the **weight** property. (which special methods do you need to add here?)

2) Add a **Table** class which is derived from the **Furniture** class.

a. Add an additional public attribute **wood** that represents the material the table is made from.
b. Add a constructor that takes two parameters: weight in lbs and wood. Should throw an exception if weight is less than or equal to 0 or wood argument is not a string (*Hint: you can use the **isinstance** built-in function to check the class of the argument to be **str**)*
c. Add a **__str__()** method that returns a string that says: "Table Item Weight: *weight* Made of: *wood.*" where the value of *weight* comes from its **weight** property and the value of *wood* comes from its **wood** property. This method should use the string returned by the base class **__str__** method.

3) Add a **Bed** class which is also derived from the Furniture class.
   a. Add a public attribute **size** that represents the size of the bed. Allowed values: "Twin", "Full", "Queen", and "King"
   b. Add a constructor that takes two parameters: weight in lbs and size. The constructor should throw an exception if weight is less than or equal to 0 or size argument is not one of the allowed values.
   c. Add a **__str__()** method returns a string that says: "Bed Item Weight: *weight* Size: size" where the value of weight comes from its **weight** property and the value of size comes from its **size** property. This method should use the string returned by the base class **__str__** method.
4) Add a **FurnitureGallery** class.
   a. Add a private attribute called **__furnList** which is a list of **Furniture** class (and derived classes) objects.
   b. Add a constructor with no arguments that initializes the **__furnList** to empty list.
   c. Add a method called **addFurniture** that takes an argument which is supposed to be a Furniture class (or derived class) object. Method should verify that the argument is one of the Furniture or derived classes and add it to **__furnList.**
   d. Add a **sort** method that sorts the **__furnList** list based on weight.
   e. Make this class iterable so that it can be used in a for-loop to iterate over the objects in the list. (What special method do you need to add here?)
5) Write all the four the classes in a single file called furniture.py. Now run the provided furnitureClient.py file. It should run without errors and should give the output as shown in the screenshot of the sample run below. ***NOTE: you shouldn't modify the client code. It just has to run correctly and give output as shown in the screenshot below.***




Submit the program in a file named first_last_furniture.py. Make sure to add a block-comment at the start of the file that lists assignment title, class name, date, your name, and assignment description. Follow naming conventions.