1.	**Problem 1** Write a generic List with a type parameter T. The class store and manipulate elements of various types. Implementation should have the following: **Class name**: `MyList<T>` **Supported methods**: `void addItem(T elt)` This method adds an item of type T to the list at the end. `T getItem(int index)` T: This method retrieves and returns the item at a specified index. `T removeItem(int index)` T: This method removes and returns the item at a specified index and shifts the remaining items accordingly. `int size()` This method returns the current number of elements in the list. `bool isEmpty()` This method checks if the list is empty. `void clear()`This method removes all items from the list. `bool contains(T elt)`This method checks whether the list contains a specific item. `T[] toArray()`This method returns an array containing all the items in the list. Now, create a class Person with name, address, and age. Now, inherit the person class and create a student class that has a university name, and a study method. Utilize the previously created MyList generic class. Store 3 person data, and store 2 student data in the list. **Problem 2** create a Shape class. Extend it in 2DShape, 3DShape class. Inherit the 2DShape class to the Circle, Rectangle, and Triangle classes. Inherit the 3DShape class to Cube, and Cylinder classes. Create a generic class Node that can take any kind of Shape. 1. write 5 assignment statements by creating instances of those classes and illustrate which assignment is valid and which is not. Explain why that is not valid by writing a comment. 2. Create a canvas class that has a method addShape(). Use wild card with bounded parameters i.e., extends and super. Now, try to invoke the method from the main code with different types of arguments. Explain which invocation is not possible and why.	**10 + 10**