

Programming Assignment 5

Basic Generics 101

Max Points 100

Due: 4/17/2022 at 11:59pm

Background Story of this Assignment

You have learned about generics in object oriented programming. This assignment will allow you to apply generics with data structures.

Start Early and see the TAs and ULAs! They are here to help you! Don't procrastinate!

Assignment Details

For this assignment, you are going to implement a generic ADT LinkedListClass. The idea is you will create one class that can handle different data types.

The Classes Name

You are going to implement 2 classes in the same Java file.

1. The first class is the Generic `Node` class with type variable `T`.

Node Class Attributes

- A Node reference called `next` that represents the reference to the next node in the ADT LinkedList.
- A primitive `T` data value called `data` that stores information in the actual node.

Node Class Constructors

The Generic Node Class has two constructors.

- Default constructor that sets all attributes to null values. When the constructor is invoked, the message "Node() Constructor Invoked..." is displayed.
- Overloaded constructor that takes one argument of type `T`. The constructor sets the data attribute to the value that was passed in the argument. The next attribute is just set to null. When the constructor is invoked, the message "Node(T data) Constructor Invoked..." is displayed.

2. The second class is the Generic `LinkedList` class with type variable `T`.

LinkedList Class Attributes

- A Node reference called `head` that represents the reference to the first node in the ADT `LinkedList`.
- A primitive `int` data value called `length` that stores the number of nodes in the linked list. The default value should be set to 0.

LinkedList Class Constructor

The Generic `LinkedList` Class has one constructors.

- Default constructor that sets all attributes to null values. When the constructor is invoked, the message “`LinkedList()` Constructor Invoked...” is displayed.

LinkedList Class Methods

The Generic `LinkedList` Class has the following non-static methods.

- A method called `insert` that takes one argument of type `T`. This method inserts the node with the data of type `T` in the back of a linked list.
- A method called `insert` that takes two arguments. The first argument is a primitive integer called `position`. This determines where in the linked list the node will be inserted. The second argument is of type `T` that represents the data being inserted into the Linked List. This method inserts the node with the data of type `T` in a certain position of the `LinkedList`. If the position passed is not within range, have the message “Out of range!” be displayed and have the method terminate.
- A method called `remove` that takes one argument. The argument is a generic type `T` variable that represents the data. The method removes a node that contains the data passed. Assume that the linked list doesn’t have duplicates.
- A method called `clear` that takes no arguments. The method removes all nodes in the linked list.
- A method called `empty` that takes no arguments. The method determines if a linked list is empty or not. The method returns a primitive Boolean value. True if the linked list is empty and false otherwise.
- A method called `length` that takes no arguments. The method determines the number of nodes in the list and returns the number.

- A method called `toString` that takes no arguments. The method will display the contents of the linked list in the form of one. Here is an example `"Sonic ---> Tails ---> Knuckles ---> Eggman"` If the list is empty, then the message `"Empty List"` is displayed.

The Provided Files

You were provided some files to assist you in this assignment.

1. A python script that will test and verify that your code output is correct.
2. A runner class that contains the main method.

Requirements

Your program must follow these requirements.

- **DO NOT CREATE YOUR OWN MAIN METHOD! I have provided a runner file for you that you will use to test your code. The graders will also be using a similar runner file to test your code as well. If you create your own main method and submit it, you will lose points in the respective categories of the rubric it affects.**
- **Name your java file `LinkedList.java`.**
- Do not add any additional attributes to the classes that were describe to you. This will result in point deductions.
- Make your attributes private!
- The output must match exactly (this includes case sensitivity, white space, and even new lines). Any differences in the output will cause the grader script to say the output is not correct. Test with the script provided in order to receive potential full credit. Points will be deducted! Check out the sample text file.
- Do not change the method signatures. Any changes to the method signatures will result in points being deducted.
- You are welcome to create additional helper methods as long as you do not remove the required methods that have been asked in this assignment.
- Do not make any changes to the Runner file that was provided for you. Any changes will result in points being deducted.
- Your code must work on Eustis. If it does not work on Eustis, points will be deducted and not changed as mentioned previously.
- **If you do not use generics, them a score of 0 will be applied to assignment overall!**
- Make sure you include a comment header. Check the assignment page of how it should. **It should be exactly the first line of your Java source file. See the assignment page for more info.**

The Rubric

Please see the assignment page in Webcourses for the Rubric of how the assignment will be evaluated.

Testing the Solution with the Python Script

Once you have completed the assignment, you will need to test it to make sure it matches Dr. Steinberg's sample output. In Eustis, make sure to upload the Python script, your Java Solution, the Runner file, and sample text output. I would highly recommend that you have a folder with those files only.

Once all of those files are uploaded into a directory in Eustis, run the command "`python3 p5testscript.py`". The script will compile your Java source and execute it. You will then see the result in the form of a happy face or sad face. The happy face means your output was correct. The sad face means something was off with the output. Remember, the script is very picky with white space and new lines. Make sure you do not add any extra trailing white space or new lines. Look at the sample text output.

Tips in Being Successful

Here are some tips and tricks that will help you with this assignment and make the experience enjoyable.

- Do not try to write out all the code and build it at the end to find syntax errors. For each new line of code written (my rule of thumb is 2-3 lines), build it to see if it compiles successfully. **It will go a long way!**
- After any successful build, run the code to see what happens and what current state you are at with the program writing so you know what to do next! If the program performs what you expected, you can then move onto the next step of the code writing. If you try to write everything at once and build it successfully to find out it doesn't work properly, you will get frustrated trying find out the logical error in your code! **Remember, logical errors are the hardest to fix and identify in a program!**
- Start the assignment early! Do not wait last minute (the day of) to begin the assignment.
- Ask questions! It's ok to ask questions. If there are any clarifications needed, please ask TAs/ULAs and the Instructor! We are here to help!!! You can also utilize the discussion board on Webcourses to share a general question about the program as long as it doesn't violate the academic dishonesty policy.