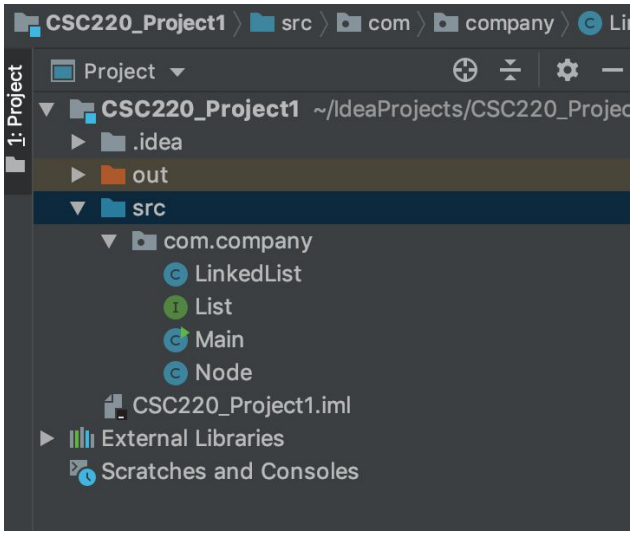
# Project 1: Calculator History

Due: 24 February 2019

In the last two lectures we explored the Linked List data structure and examined its usefulness as a flexible storage container. An implementation of the Linked List is provided in this project. Before beginning, review the code in the List, LinkedList, and Node .java files.

Once you have reviewed the source code we have been using in class, create a new project in your IDE. Find the folder on your computer containing your project's source code. Copy the provided List.java, LinkedList.java, and Node.java files into this directory. If you have copied them into the correct folder, the List interface, LinkedList class, and Node class should appear in your IDE alongside your Main class in the panel on the left.




The goal for Project 1 is to construct a calculator program which stores the user's past answers in a Linked List. The calculator should support addition, subtraction, multiplication, and division operations. When a calculation has been completed, append the answer to the Linked List.

**To use the Linked List we have constructed in class, define a variable with the data type "LinkedList". Refer to the LinkedList.java source code as needed.**

Your program should begin by asking the user which operation they would like to perform:

```
1) Addition
2) Subtraction
3) Multiplication
4) Division
5) See History
```

Allow the user to select an option. If option 1 - 4 is selected, allow the user to enter two numbers to be used in the relevant calculation. When computing the result, output the answer on the screen and also store the output in the Linked List you created. Hint: we want to **append** a new Node to the linked list.

If option 5 is selected, display the contents of the Linked List.

**Example** (user input is shown here in **bold** text):

```
Select an option:
1) Addition
2) Subtraction
3) Multiplication
4) Division
5) See History

Option: 1

Performing Addition.
Enter first number: 5
Enter second number: 7

5 + 2 = 7


Select an option:
1) Addition
2) Subtraction
3) Multiplication
4) Division
5) See History

Option: 4

Performing Division.
Enter first number: 9
```

```
      Enter second number: 3

      9 / 3 = 3

Select an option:
      1) Addition
      2) Subtraction
      3) Multiplication
      4) Division
      5) See History

      Option: 5

      History:
      7
      3
```

**Submission:**
Submit all **.java** files used in your program.