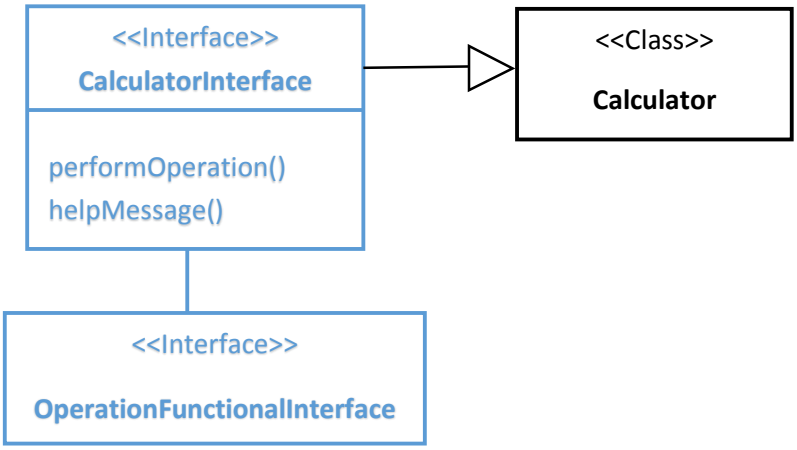
Week 10: Lambda Expressions

Assignment 1: Calculator.java

In this problem we had to implement the basic working of a calculator with the help of lambda expressions. Our initial approach was to create individual methods for each of the operations and to pass the arguments achieved by the user to that method for performing that operation. But this proved to be an auxiliary task, which could be further drilled down.

Eventually, we implemented the operations by directly passing the lambda expressions in the print statements which would directly return the results of the required operation. We had been provided supplementary interfaces which simplified our problem implementation furthermore.



Assignment 2: Elevator.java

This week's problem asked us to modify our last week's assignment of the Elevator problem using lambda expressions. Last week's implementation had a few design issues which after taking inputs from the Professor we tried to incorporate those changes in our implementation.

- We have implemented our code in such a way that every time the program starts the person thread should start as well.
- It should also include different functions which defined actions like person going to the office, entering the elevator, getting down from the elevator etc. For this we have defined states for a person for the current actions they are performing.
- We have – as per suggestion – used a Queue for each floor, to accommodate the information of each individual person going to or from that floor.
- We have performed for the ease of implementation, sorting, based on the number of floor which each person had to get down, this helped us to traverse the elevator in a much simpler manner.
- Initially we had just implemented an Elevator class and a Person class, but as per the suggestion professor made, we have also implemented a driver class which actually controls the main working of our supporting classes.

The final implementation saw us incorporating lambdas with our renewed implementation of the classes. In addition to that we had to give summary of each individual regarding the weight statistics which they are contributing towards measuring the current capacity of the elevator. We have used lambdas to calculate this by implementing interfaces required to get those results. By using the *filter* and *forEach* method we are printing the stats for every even numbered user for the elevator. We have also used lambdas for creation for each individual person.

Improved flow for the implementation: