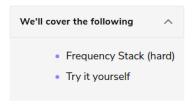




Problem Challenge 3



Frequency Stack (hard)

Design a class that simulates a Stack data structure, implementing the following two operations:

- 1. push(int num): Pushes the number 'num' on the stack.
- 2. pop(): Returns the most frequent number in the stack. If there is a tie, return the number which was pushed later.

Example:

```
After following push operations: push(1), push(2), push(3), push(2), push(1), push(2), push(5)

1. pop() should return 2, as it is the most frequent number

2. Next pop() should return 1

3. Next pop() should return 2
```

Try it yourself

Try solving this question here:

```
Python3
👙 Java
                                   ⓒ C++
     lass FrequencyStack:
      def push(self, num):
       return 0
      def pop(self):
     frequencyStack()
      frequencyStack.push(1)
      frequencyStack.push(2)
      frequencyStack.push(3)
      frequencyStack.push(2)
     frequencyStack.push(1)
      frequencyStack.push(2)
      frequencyStack.push(5)
      print(frequencyStack.pop())
      print(frequencyStack.pop())
      print(frequencyStack.pop())
Run
```







Report an Issue Ask a Question

