Interview Cake

← course home (/table-of-contents#section_queues-stacks_question_largest-stack)

You want to be able to access the *largest element* in a stack.

You've already implemented this `Stack` class:

Python 2.7

```
class Stack(object):

    def __init__(self):
        """Initialize an empty stack"""
        self.items = []

    def push(self, item):
        """Push a new item onto the stack"""
        self.items.append(item)

    def pop(self):
        """Remove and return the last item"""
        # If the stack is empty, return None
        # (it would also be reasonable to throw an exception)
        if not self.items:
            return None

        return self.items.pop()

    def peek(self):
        """Return the last item without removing it"""
        if not self.items:
            return None
        return self.items[-1]
```

Use your `Stack` class to **implement a *new* class `MaxStack` with a method `get_max()` that returns the largest element in the stack.** `get_max()` should not remove the item.

Your stacks will contain only integers.
