In Python, it is easy to build stacks because we have got Lists built-in, which are arrays in their essence. Lists also have methods like append() and pop() that do the same as push() and pop() in stacks. Therefore, implementing stacks in Python is quite simple.

In [4]:

```
class Stack():
    def __init__(self):
        self.array = []
# We start building our Stack by creating a constructor that will contain an empty list(array).
# There is no need to create length, because this comes built-in with Lists.

    def peek(self):
        return self.array[len(self.array) - 1]
# The peek method accesses the last element of the array(top element of the stack) by using the built-in length fu

    def push(self, data):
        self.array.append(data)
        return
# The push method inserts an element at the end of the list(top of the stack) using append(), built-in with Lists.

    def pop(self):
        if len(self.array) != 0:
            self.array.pop()
            return
        else:
            print('Oops! This stack is empty.')
            return
# The pop method will remove the last element of the list(top element of the stack) using pop(), built-in with Lis
# They time complexity of this operation for the last element of the list is O(1).

    def get_stack(self):
        for i in range(len(self.array) -1, -1, -1):
            print(self.array[i])
        return
# As stacks follow the LIFO principle (Last In-First Out), the method needs to output the last item of the list fi
# This will require loop traversing through the list, which will affect the time complexity of this operation: O(n
```

All that is left to do now is testing our blueprint:

In [5]:
```
my_library = Stack()

my_library.push("The Little Prince")
my_library.push("Harry Potter and the Philosopher's Stone")
my_library.push("The Gruffalo")
my_library.push("The Tale of Peter Rabbit")

my_library.get_stack()
```

```
The Tale of Peter Rabbit
The Gruffalo
Harry Potter and the Philosopher's Stone
The Little Prince
```

Here we can see the LIFO principle in action. While the first item added to our stack was 'The Little Prince', the first one to be output was 'The Tale of Peter Rabbit').

In [6]:
```
my_library.pop()
my_library.pop()

my_library.get_stack()
```

```
Harry Potter and the Philosopher's Stone
The Little Prince
```

In [9]:
```
print(my_library.peek())
```

```
Harry Potter and the Philosopher's Stone
```

In [10]:
```
print(my_library.__dict__)
```

```
{'array': ['The Little Prince', "Harry Potter and the Philosopher's Stone"]}
```