Methods and attributes

Stack class

A stack is a data structure where elements are stored in the order in which they were added. A stack is like a stack of books or plates: you cannot remove any object, only the one that was added last.

Implement your Stack class in Python. It should wrap several built-in Python data structure and implement the following methods:

- `push(el)` : push a new element `el` to the stack;
- `pop()` : remove and return the last element from the stack;
- `peek()` : return the last element without removing it;
- `is_empty()` : check if the stack is empty and return `True` or `False` .

The stack must be empty upon initialization. You don't need to create an instance of the class or print anything.

Source: JetBrains Academy

Solution

In [137…

```
class Stack:
    def __init__(self):
        self.stack = []

    def push(self, el):
        self.stack.append(el)

    def pop(self):
        if not self.is_empty():
            return self.stack.pop()
        else:
            print("Stack is empty!")
            return None

    def peek(self):
        if not self.is_empty():
            return self.stack[-1]
        else:
            print("Stack is empty!")
            return None

    def is_empty(self):
        return len(self.stack) == 0
```

In [138…

```
# Create an empty stack
my_stack = Stack()
```

```
# Check if the stack is empty
print(my_stack.is_empty())
```

```
True
```

In [139…
```
# Push some elements on the stack
my_stack.push(0)
my_stack.push(1)
my_stack.push(2)
my_stack.push(3)
my_stack.push(4)

# Check again if the stack is empty
print(my_stack.is_empty())
```

```
False
```

In [140…
```
# Remove some elements from the stack
print(my_stack.pop())
print(my_stack.pop())
```

```
4
3
```

In [141…
```
# Get the last element from the stack
print(my_stack.peek())
```

```
2
```

In [142…
```
# Remove more elements from the stack
print(my_stack.pop())
print(my_stack.pop())
print(my_stack.pop())
```

```
2
1
0
```

In [143…
```
# Check again if the stack is empty
print(my_stack.is_empty())
```

```
True
```

In [144…
```
# Try to remove one more element from the stack
print(my_stack.pop())
```

```
Stack is empty!
None
```

In [ ]:
```

```