Python - Stack

In the english dictionary the word stack means arranging objects on over another. It is the same way memory is allocated in this data structure. It stores the data elements in a similar fashion as a bunch of plates are stored one above another in the kitchen. So stack data strcuture allows operations at one end wich can be called top of the stack.We can add elements or remove elements only form this en dof the stack.

In a stack the element insreted last in sequence will come out first as we can remove only from the top of the stack. Such feature is known as Last in First Out(LIFO) feature. The operations of adding and removing the elements is known as **PUSH** and **POP**. In the following program we implement it as **add** and and **remove** functions. We declare an empty list and use the append() and pop() methods to add and remove the data elements.

PUSH into a Stack

Let us understand, how to use PUSH in Stack. Refer the program mentioned program below −

Example

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

   def add(self, dataval):
# Use list append method to add element
```

```
        if dataval not in self.stack:
           self.stack.append(dataval)
           return True
        else:
           return False
  # Use peek to look at the top of the stack
     def peek(self):
            return self.stack[-1]

  AStack = Stack()
  AStack.add("Mon")
  AStack.add("Tue")
  AStack.peek()
  print(AStack.peek())
  AStack.add("Wed")
  AStack.add("Thu")
  print(AStack.peek())
```

Output

When the above code is executed, it produces the following result −

```
Tue
Thu
```

POP from a Stack

As we know we can remove only the top most data element from the stack, we implement a python program which does that. The remove function in the following program returns the top most element. we check the top element by calculating the size of the stack first and then use the in-built pop() method to find out the top most element.

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

   def add(self, dataval):
# Use list append method to add element
      if dataval not in self.stack:
         self.stack.append(dataval)
         return True
      else:
         return False

# Use list pop method to remove element
   def remove(self):
      if len(self.stack) <= 0:
         return ("No element in the Stack")
      else:
         return self.stack.pop()

AStack = Stack()
AStack.add("Mon")
AStack.add("Tue")
AStack.add("Wed")
AStack.add("Thu")
print(AStack.remove())
print(AStack.remove())
```

Output

When the above code is executed, it produces the following result −

```
Thu
Wed
```

Print Page