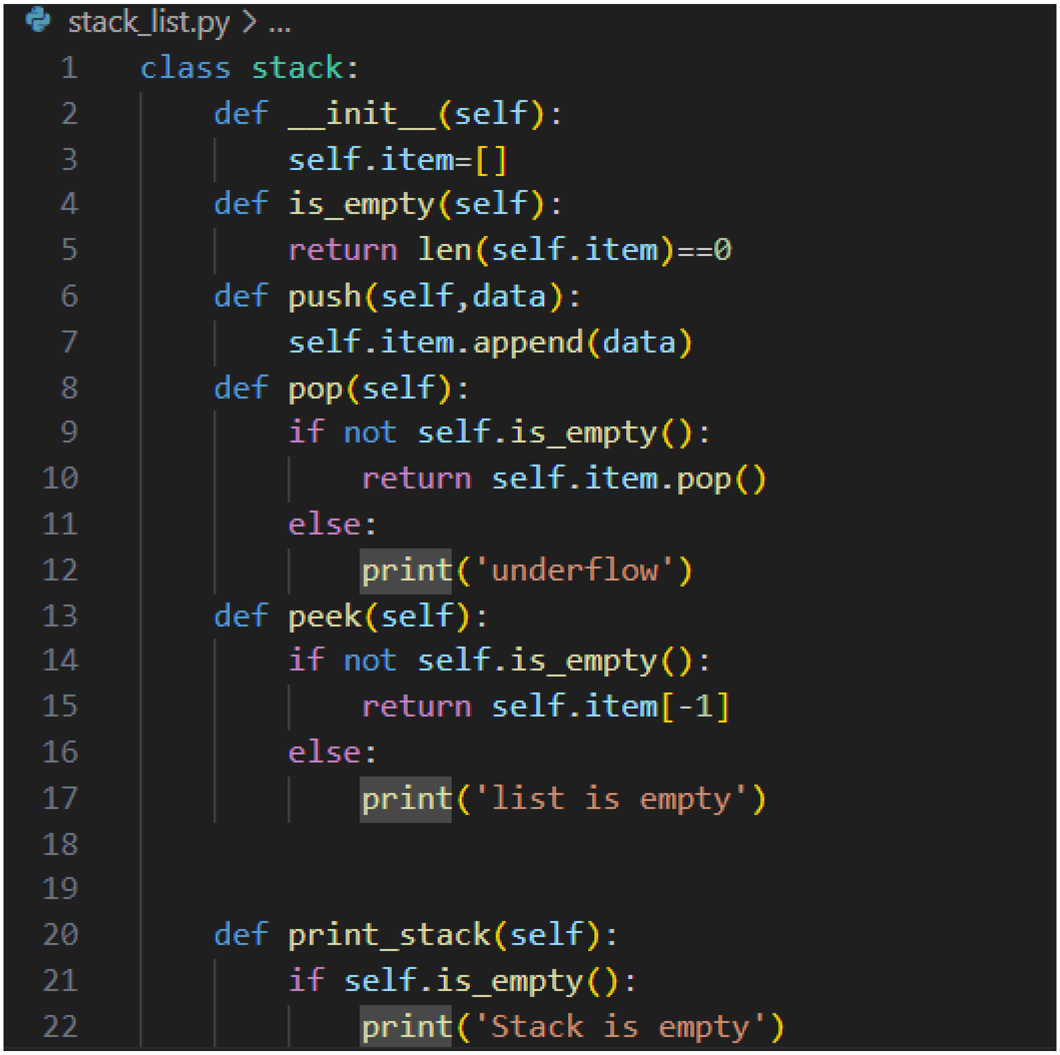
Lab#8

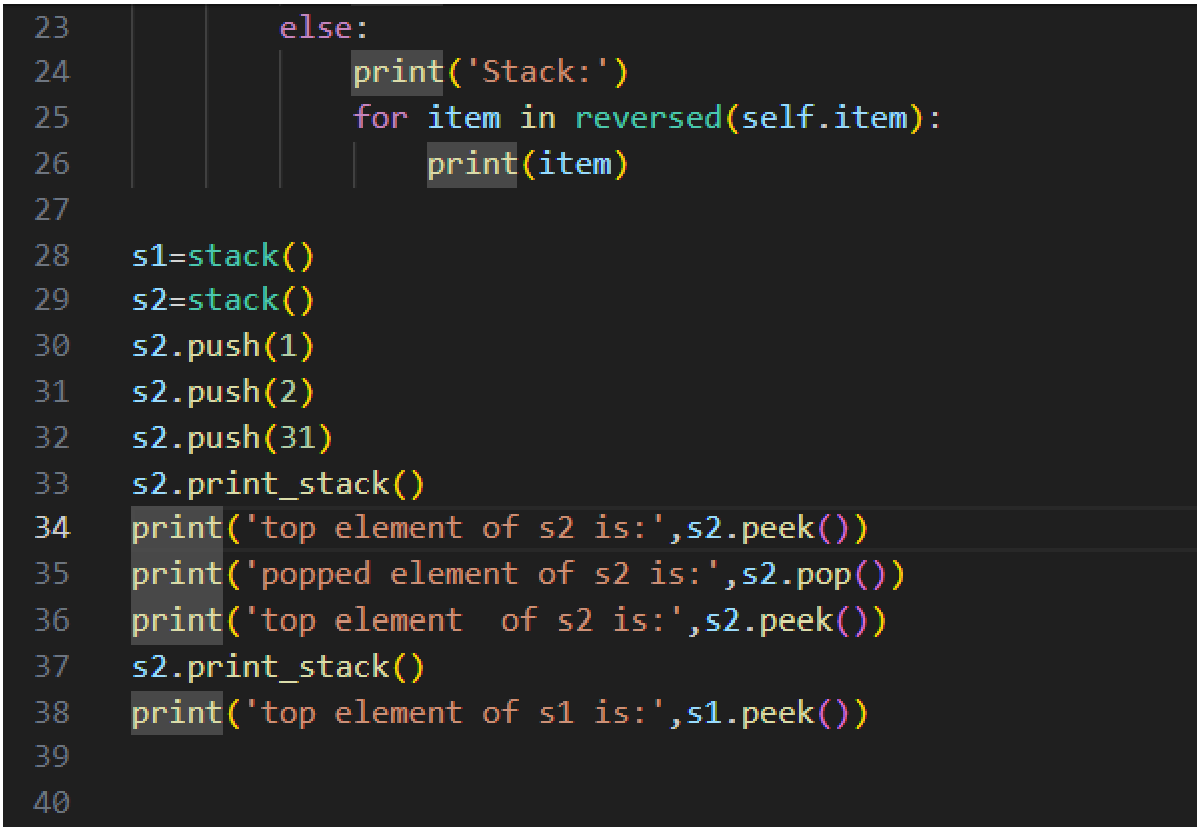
Q.1: Write a program to create a stack data structure by using a list.

Solution:



```python
class stack:
    def __init__(self):
        self.item=[]
    def is_empty(self):
        return len(self.item)==0
    def push(self,data):
        self.item.append(data)
    def pop(self):
        if not self.is_empty():
            return self.item.pop()
        else:
            print('underflow')
    def peek(self):
        if not self.is_empty():
            return self.item[-1]
        else:
            print('list is empty')


    def print_stack(self):
        if self.is_empty():
            print('Stack is empty')
```

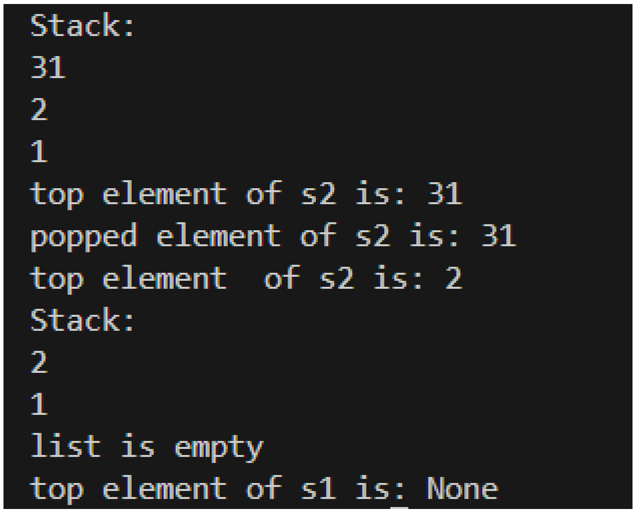


```python
        else:
            print('Stack:')
            for item in reversed(self.item):
                print(item)

s1=stack()
s2=stack()
s2.push(1)
s2.push(2)
s2.push(31)
s2.print_stack()
print('top element of s2 is:',s2.peek())
print('popped element of s2 is:',s2.pop())
print('top element  of s2 is:',s2.peek())
s2.print_stack()
print('top element of s1 is:',s1.peek())


```

Result:



```
Stack:
31
2
1
top element of s2 is: 31
popped element of s2 is: 31
top element  of s2 is: 2
Stack:
2
1
list is empty
top element of s1 is: None
```

Class Assignment

Q: Modify the above program example#1 lab#8 by creating a stack using singly linked list data structure.