In [1]:
```
from queue import LifoQueue as Stack
from queue import Queue
```

In [2]:
```
def peek(obj):
    if type(obj) == Queue: return obj.queue[0]
    elif type(obj) == Stack: return obj.queue[-1]
```

In [3]:
```
def display(obj):
    print('[ ',end='')
    if type(obj) == Queue:
        for i in range(len(obj.queue)):
            print(obj.queue[i],end=' ')
        print(']')
    elif type(obj) == Stack:
        for i in range(len(obj.queue)):
            print(obj.queue[-(i+1)],end=' ')
        print(']')
```

In [4]:
```
stack = Stack()
stack.put(2)
stack.put(5)
stack.put(8)
stack.put(1)
stack.put(9)
```

In [5]:
```
peek(stack)
```

Out[5]:
```
9
```

In [6]:
```
stack.qsize()
```

Out[6]:
```
5
```

In [7]:
```
display(stack)
```

```
[ 9 1 8 5 2 ]
```

In [8]:
```
queue = Queue()
queue.put(2)
queue.put(8)
queue.put(4)
queue.put(5)
queue.put(1)
```

In [9]:
```
peek(queue)
```

Out[9]:
```
2
```

In [10]:
```
queue.qsize()
```

Out[10]:
```
5
```

In [11]:
```
display(queue)
```

```
[ 2 8 4 5 1 ]
```