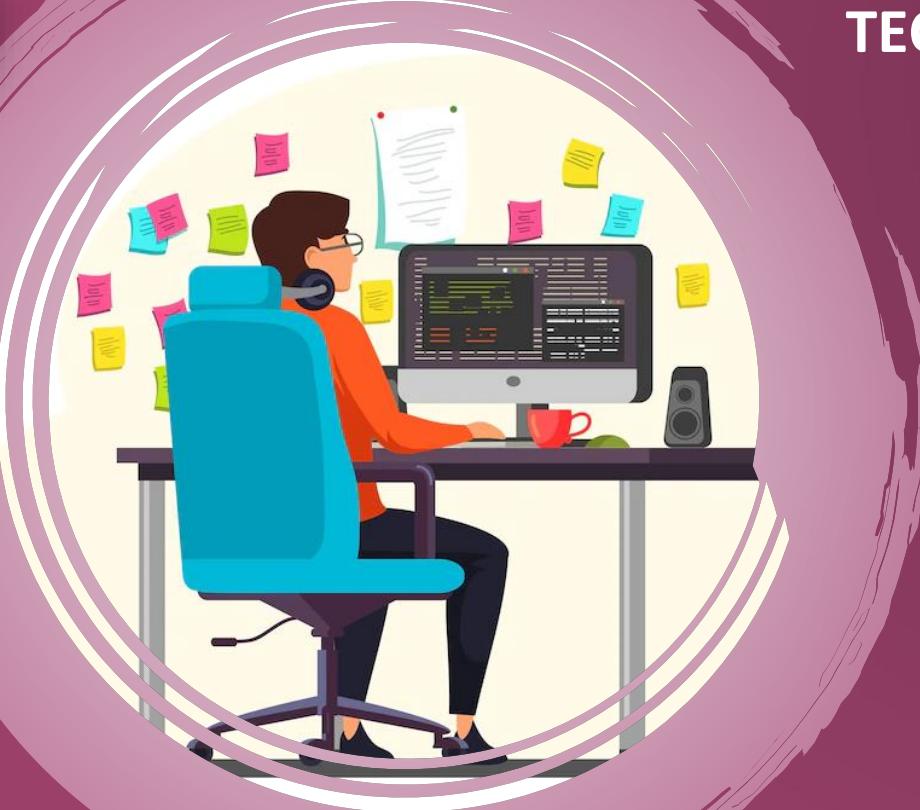




TAMILNADU ADVANCED
TECHNICAL TRAINING INSTITUTE



PYTHON

Queue

Queue in Python is a linear data structure with a rear and a front end, similar to a stack. It stores items sequentially in a FIFO (First In First Out) manner.

Operations

- Enqueue
- Dequeue
- Front
- Rear
- isEmpty
- isFull

Types

- Input Restricted Queue
- Output Restricted Queue
- Circular Queue
- Double-Ended Queue (Dequeue)
- Priority Queue

Implementation

- list
- collections.deque
- queue.Queue

Programs

```
queue = []  
  
queue.append('a')  
  
queue.append('b')  
  
queue.append('c')  
  
print(queue)  
  
print(queue.pop(0))  
  
print(queue.pop(0))  
  
print(queue)
```

Programs

```
from collections import deque
```

```
q = deque()
```

```
q.append('a')
```

```
q.append('b')
```

```
q.append('c')
```

```
print(q.popleft())
```

```
print(q.popleft())
```

```
print(q)
```

Programs

```
from queue import Queue
```

```
q = Queue(maxsize = 3)
```

```
print(q.qsize())
```

```
q.put('a')
```

```
q.put('b')
```

```
print("\nFull: ", q.full())
```

```
print(q.get())
```

```
print(q.get())
```

```
print("\nEmpty: ", q.empty())
```

Programs

```
pri_que = []  
  
pri_que.append((2, 'A'))  
  
pri_que.append((1, 'B'))  
  
pri_que.append((3, 'C'))  
  
pri_que.sort(reverse=True)  
  
while pri_que:  
  
    next_item = pri_que.pop()  
  
    print(next_item)
```



Programs

```
from queue import PriorityQueue  
  
q = PriorityQueue()  
  
q.put((2, Python))  
  
q.put((1, Java))  
  
q.put((3, C))  
  
while not q.empty():  
  
    next_item = q.get()  
  
    print(next_item)
```