Queue (stl)

- queue is linear data structure. operations on quee are performed in FIFO (First in first out) principle.

→ To use queue as STL, we need to first include queue library –

```
#include<queue>
```

→ How to declare a queue –

```
queue < int > q;
```

keyword   Datatype   variable

- operations on queue

1.) push() / emplace() :

- push() is used for insert the element in queue.
- you can use emplace() instead of push()
- The main difference between push() and emplace() is – emplace() is faster and cheaper than push().

eg. For push ()

```
q.push(7); // {7}
q.push(9); // {7,9}
q.push(2); // {7,9,2}
q.emplace(5); // {7,9,2,5}.
```

2.> ~~top()~~ front ()

- Access the first element in queve.

```
cout << q.front(); // output : 7
```

3.> pop ()

- Remore the first element of queve.

```
cout << q.front(); // output : 7
pop();
cout << q.front(); // output : 9
```

4.> back()

- Access the last element of queve.

```
cout << q.back(); // output : 5
```

5.> size()

- It will give the size of queue

```
// {9, 2, 5}
cout << q.size(); //output : 3
```

6.> empty()

- It will return true if the queue will empty else it will return false.

```
cout << q.empty(); // output : 0
```