Day 26

24/03/2025

Hashtable :- (key, value); (unordered)

Hashtable HT = New Hashtable();

To set capacity

HT.Add("1", "Hello");

"1" → key

"Hello" → value.

Should be unique.

HT.containskey("1") -> TOCheck

HT.containsValue("Hello") -> TOCheck.

String value = HT["1"].ToString();

HT.Remove("1");

"1" → key

HT.Clear() -> clear All the value.

Foreach (object item in HT.Keys)

↓

To get key.

Foreach (object item in HT.Values)

↓

To get value.

Foreach (DictionaryEntry item in HT)

↓

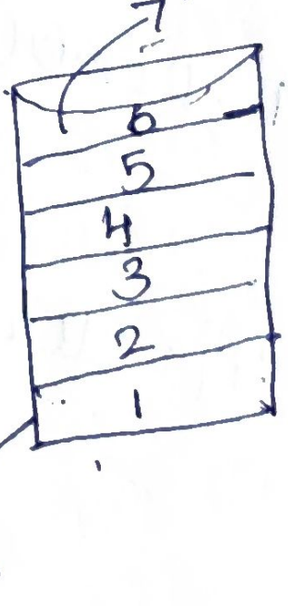
To get both value and key

item.key.ToString();

item.value.ToString();

Stack : (Last in first out)

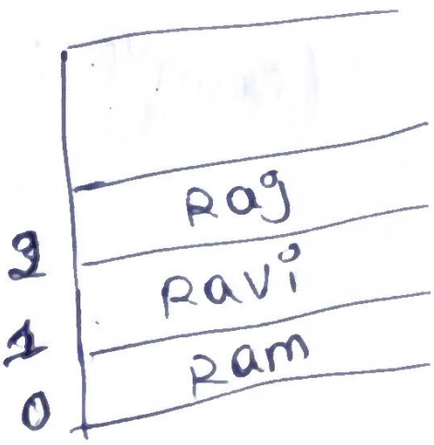
Example: To find new employee join the company).



Storage of Stack

Stack Sk = New Stack();

Sk.push("Ram");
Sk.push("Ravi");
Sk.push("Raj");



Sk.peek();

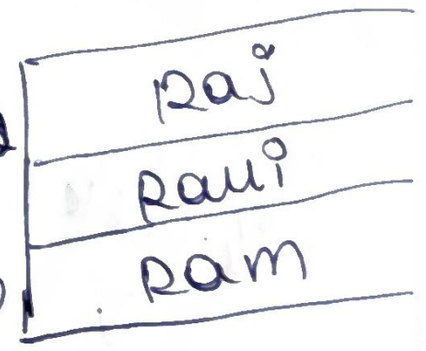
↓

it will show the last value we given.

Sk.POP(); → Show and delete

↓

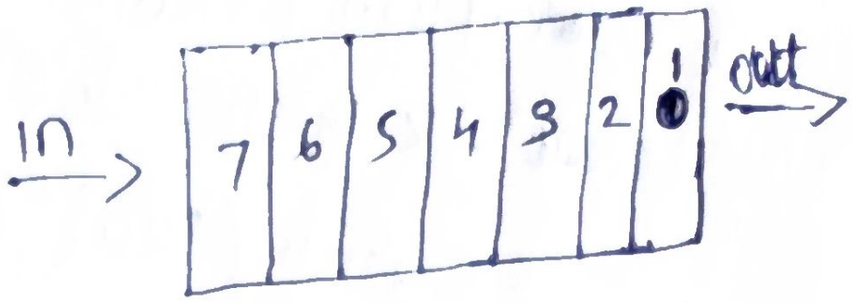
it will delete the last value we given.



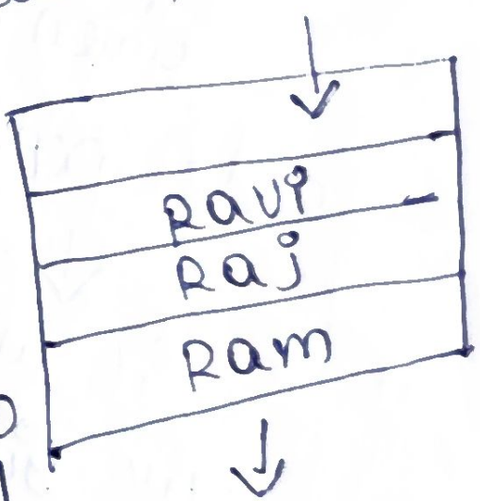
It will delete Raj first, next Ravi and then Ram.

Queue :- (First in First out)

(example : Patient TOKEN system)




```
Queue q = new queue();
    q.Enqueue("Ram");
    q.Enqueue("Raj");
    q.Enqueue("Ravi");
```



q.peek(); → 0

↓

It will show Ram

First.

q.Dequeue(); → 0 → it will delete Ram

list.ItemSources = null; → clear and

list.ItemSources = q; → Add

↓

It will show the list.