

Stack__7

```
} <public interface Stack<E_1
```

```
boolean isEmpty // التحقق مما إذا كان المكس فارغًا
```

```
int size // إرجاع عدد العناصر في المكس
```

```
E top // إرجاع العنصر العلوي بدون حذفه
```

```
void push(E e) // إضافة عنصر إلى المكس
```

```
E pop // إزالة وإرجاع العنصر العلوي
```

```
{
```

```
} <public class ArrayStack<E> implements Stack<E__2
```

```
private E[] data
```

```
private int topIndex // يشير إلى أعلى عنصر في المكس
```

```
public ArrayStack(int capacity)
```

```
data = (E[]) new Object[capacity]
```

```
topIndex = -1
```

```
{
```

```
Override@
```

```
public boolean isEmpty
```

```
return topIndex == -1
```

```
{
```

```
Override@
```

```
public int size
```

```
return topIndex + 1
```

```
{
```

```
Override@
```

```
public E top
```

Stack__7

```
        } if (isEmpty())
;("المكدس فارغ")throw new IllegalStateException
        {
            ;return data[topIndex]
        }

@Override
        } public void push(E e)
        { if (size() == data.length)
;("المكدس ممتلئ")throw new IllegalStateException
        {
            ;data[++topIndex] = e
        }

@Override
        } ()public E pop
        { if (isEmpty())
;("المكدس فارغ")throw new IllegalStateException
        {
            ;E result = data[topIndex]
;data[topIndex--] = null // إزالة العنصر وتحرير الذاكرة
            ;return result
        }
    }
```

Stack__7

```
} <public class LinkedStack<E> implements Stack<E> {  
    } <private static class Node<E> {  
        ;E element  
        ;Node<E> next  
  
    } Node(E element, Node<E> next)  
        ;this.element = element  
        ;this.next = next  
    }  
  
    ;private Node<E> topNode  
    ;private int size  
  
    } ()public LinkedStack  
        ;topNode = null  
        ;size = 0  
    {  
  
        Override@  
    } ()public boolean isEmpty  
        ;return topNode == null  
    {
```

Stack__7

```
        Override@
    } ()public int size
        ;return size
        {

        Override@
    } ()public E top
    } if (isEmpty())
;("المكدس فارغ")throw new IllegalStateException
        {
            ;return topNode.element
        {

        Override@
    } public void push(E e)
;topNode = new Node<>(e, topNode)
        ;++size
        {

        Override@
    } ()public E pop
    } if (isEmpty())
;("المكدس فارغ")throw new IllegalStateException
```

Stack__7

```
        {
            ;E result = topNode.element
            ;topNode = topNode.next
            ;--size
            ;return result
        }
    {
        isEmpty__٤
        Override@
    } ()public boolean isEmpty
        ;return topIndex == -1
    {

        size()__٥
        Override@
    } ()public int size
        ;return topIndex + 1
    {
        top()
        Override@
    } ()public E top
        } if (isEmpty())
        ;("المكدس فارغ")throw new IllegalStateException
    {
        ;return data[topIndex]
```

Stack__7

```
{
    ٧.دالة push(E e)
        @Override
        public void push(E e)
        {
            if (size() == data.length)
                throw new IllegalStateException("المكدس ممتلئ");
        }
        ;data[++topIndex] = e
    }
    5.دالة pop()
        @Override
        public E pop()
        {
            if (isEmpty())
                throw new IllegalStateException("المكدس فارغ");
            {
                ;E result = data[topIndex]
                ;data[topIndex--] = null
                ;return result
            }
        }
    }
```