栈的实现 – 方法、函数

```java
public T push(T element) {
    if (size == capacity) {
        resize();
    }
    elementData[size++] = element;
    return element;
}
```

```java
public T peek() {
    if (this.size == 0) {
        // throw Exception
        throw new IllegalStateException();
    }
    return elementData[size - 1];
}
```

```java
public T pop() {
    if (size == 0) {
        // throw Exception
        throw new EmptyStackException();
    }

    T element = elementData[--this.size];
    return element;
}
```

```java
public boolean isEmpty() {
    return this.size == 0;
}
```

什么时候考虑使用栈

- 调用函数
- 递归
- 深度优先搜素DFS(Depth-first Search)