

```
public interface Stack<E> extends Collection<E> {
```

```
    /**
```

```
     * Tests if this stack is empty.
     * @return true if the stack is empty
     */
```

```
    public abstract boolean empty();
```

```
    /**
```

```
     * Looks at the object at the top of this stack without removing it from the stack.
     * @return the object at the top of the stack
     * @throws EmptyStackException if the stack is empty
     */
```

```
    public abstract E peek();
```

```
    /**
```

```
     * Removes the object at the top of this stack and returns that object as the value of this function.
     * @return the item at the top of the stack
     * @throws EmptyStackException if the stack is empty
     */
```

```
    public abstract E pop();
```

```
    /**
```

```
     * Pushes an item onto the top of this stack.
     * @param item to put on the top of the stack
     * @return the object passed in
     */
```

```
    public abstract E push(E item);
```

```
    /**
```

```
     * Search for an object to see if it is in the stack
     * @param
     * @return
     */
```

```
    public abstract int search(E o);
```

```
}
```