

LAZYNESS

¿En que momento se produce el proceso?

```
String todos = listaPersonas.stream()  
    .map(it -> it.getNombre())  
    .sorted( (o1, o2) -> o1.length() - o2.length())  
    .map( it -> "'" + it + "'")  
    .reduce("", (a, b) -> a + "\n" + b);
```



```
Stream<String> parcial = listaPersonas.stream()  
    .map(it -> it.getNombre() );
```

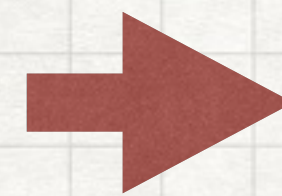
¿Se ejecuta la función? NO

El pipeline no se inicia hasta encontrar una operación terminal


```
Stream<String> parcial = listaPersonas.stream()  
    .map(it -> it.getNombre() );
```

¿Que produce map?

Stream<Persona>



Stream<String>

Problema

```
Stream<String> strings = Stream.of("b", "c", "a");  
strings.sorted();  
String p = strings.reduce("", String::concat);
```

¿Que contiene p?

abc

bca

¿Otra opción?

Problema

```
Stream<String> strings = Stream.of("b", "c", "a");  
  
strings.sorted();  
  
String p = strings.reduce("", String::concat);
```

```
Exception in thread "main" java.lang.IllegalStateException:  
    stream has already been operated upon or closed  
at java.util.stream.AbstractPipeline.evaluate(AbstractPipeline.java:229)  
at java.util.stream.ReferencePipeline.reduce(ReferencePipeline.java:474)  
at org.formacion.Main.main(Main.java:16)
```


Problema

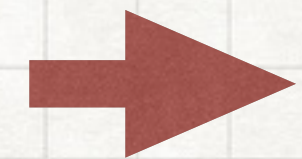
➔ `Stream<String> strings = Stream.of("b", "c", "a");`
`strings.sorted();`
`String p = strings.reduce("", String::concat);`

strings

`Stream<String>`

Inválido

`Stream<String>`



```
String p = Stream.of("b", "c", "a")  
                  .sorted()  
                  .reduce("", String::concat);
```

Stream<String>

Inválido

Stream<String>

Inválido

abc