

"And now Edgar's gone. ... Something's going on around here."

Refactoring

Liste cada uno de los cambios necesarios y la línea afectada

*Renombre variable *idx* (referenciada en línea 17) por *index*

*Renombre variable *result* (referenciada en línea 18) por *position*

```
1 package roo2;
2
3 public class CharRing extends Object {
4     char[] source;
5     int idx;
6     public class CharRing{
7         char result;
8         source = new char[srcString.length()];
9         srcString.getChars(0,srcString.length(), source, 0);
10        result = 0;
11        idx = result;
12    };
13    public char next( ){
14        char result;
15        if(idx >= source.length)
16            idx=0;
17        result = idx++;
18        return source[result];
19    };
20 }
```

Refactoring (10 minutos)

Liste cada uno de los cambios necesarios y la línea afectada

1. Renombre variable *idx* (referenciada en línea 17) por *index*
 - a. Cuál es la línea 17?
 - b. La referencia a **idx** es
 - i. lectura o escritura?
 - c. Otra línea tiene referencian a **idx**?
 - d. Dónde está declarada?
2. Renombre variable *result* (referenciada en línea 18) por *position*

```
1 package roo2;
2
3 public class CharRing extends Object {
4     char[] source;
5     int idx;
6     public class CharRing{
7         char result;
8         source = new char[srcString.length()];
9         srcString.getChars(0,srcString.length(), source, 0);
10        result = 0;
11        idx = result;
12    };
13    public char next( ){
14        char result;
15        if(idx >= source.length)
16            idx=0;
17        result = idx++;
18        return source[result];
19    };
20 }
```

Refactoring (10 minutos)

Liste cada uno de los cambios necesarios y la línea afectada

1. Renombre variable idx (referenciada en línea 17) por index
2. Renombre variable result (referenciada en línea 18) por position
 - a.Cuál es la línea 18?
 - b. La referencia a **result** es
 - i. lectura o escritura?
 - c. Otra línea tiene referencian a **result**?
 - d. Dónde está declarada?

```
1 package roo2;
2
3 public class CharRing extends Object {
4     char[] source;
5     int idx;
6     public class CharRing{
7         char result;
8         source = new char[srcString.length()];
9         srcString.getChars(0,srcString.length(), source, 0);
10        result = 0;
11        idx = result;
12    };
13    public char next( ){
14        char result;
15        if(idx >= source.length)
16            idx=0;
17        result = idx++;
18        return source[result];
19    };
20 }
```

Qué otro refactorings debería hacer para completar el ejercicio? (10 minutos)

```
1 package roo2;
2
3 public class CharRing extends Object {
4     char[] source;
5     int idx;
6     public class CharRing{
7         char result;
8         source = new char[srcString.length()];
9         srcString.getChars(0,srcString.length(), source, 0);
10        result = 0;
11        idx = result;
12    };
13    public char next( ){
14        char result;
15        if(idx >= source.length)
16            idx=0;
17        result = idx++;
18        return source[result];
19    };
20 }
```

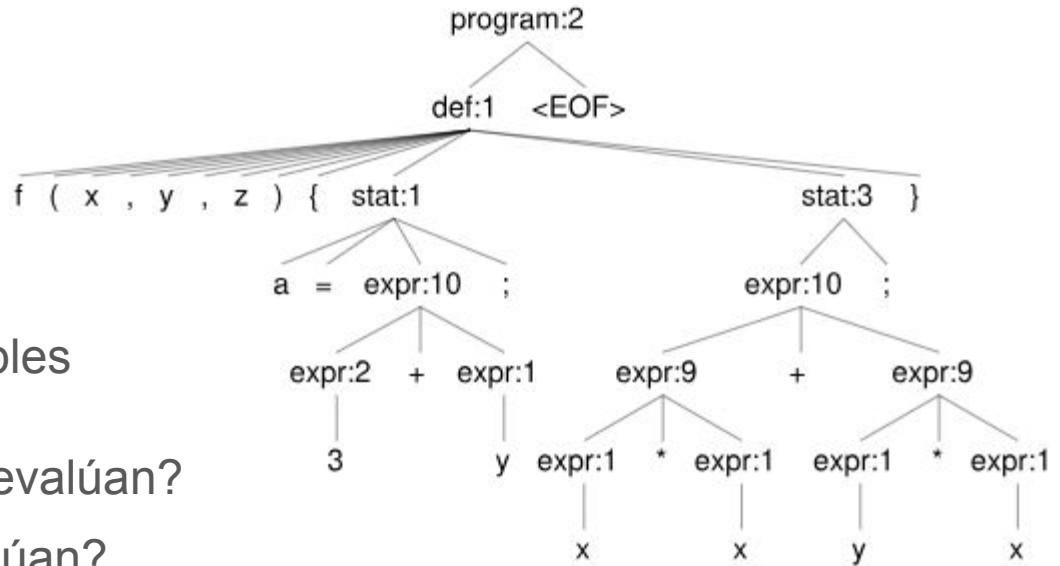
Code Smells?

```
f(x, y, z) {  
    a = 3 + y;  
    x * x + y * x;  
}
```

Declaración implícita de variables

Parámetros sin uso: cómo lo evalúan?

Variable sin uso: cómo lo evalúan?



roo2.next()

Consultas (Lifia)

1. Lunes 9: 4:30pm a 6:00pm
2. Martes 10: 5:00pm a 18:30pm
3. Jueves 12: 5:00pm a 18:30pm

Evaluación Refactoring:

Septiembre 18 Aula 3 **18hs**

