


Open · Opened 2 weeks ago by  **Rubén Montero**

# Un número aleatorio entre dos valores que definen un rango

## Resumen

- Escribiremos un método nuevo en `SimpleMathDemo` que reciba **2** valores enteros, `start` y `length`
- Devolverá un número entero **aleatorio** en el rango definido por esos valores

## Descripción

Un rango es un espacio de valores entre un mínimo y un máximo.

Por ejemplo, podemos hablar del rango de valores **entre 0 y 100**, ó **entre 56 y 205**.

```
[0, 100]
[56, 205]
```

Una forma habitual de *definir* un rango en informática es usar:

- Valor mínimo del rango (extremo izquierdo)
- Longitud del rango (tamaño)

Llámemoslos `start` y `length`.

## Un ejemplo

Para `start = 0` y `length = 10`, el rango será:

```
[0, 10]
```

Para `start = 10` y `length = 10`, el rango será:

```
[10, 20]
```

Para `start = 25` y `length = 5`, el rango será:

```
[25, 30]
```

## La tarea

**Se pide** que:

- Añadas un nuevo método `randomNumberInRange` a `SimpleMathDemo` :
  - `public`
  - Devolverá un tipo `int`
  - Recibirá (parámetros formales) **2** datos tipos `int`
    - El primero se llamará `start`
    - El segundo se llamará `length`
  - En el *cuerpo* del método, implementarás la *lógica* necesaria para devolver finalmente ( `return` ) un número entero **aleatorio** comprendido en el rango definido por `start` y `length` :
    - **Mayor o igual** que `start`
    - **Menor estrictamente** que `start+length`
  - Si detectas *algún problema* con los valores de entrada, devolverás un `-1`<sup>1</sup>

## Por ejemplo

Si desde `Main.java` invocas el método así:

```
SimpleMathDemo demo = new SimpleMathDemo();
```

```
int result = demo.randomNumberInRange(3, 5);  
// Imprimimos el valor devuelto para comprobarlo  
System.out.println(result);
```

...se espera que el valor devuelto sea 3, 4, 5, 6 ó 7, aleatoriamente.

## ¿Números aleatorios?

Puedes generar un número aleatorio *decimal* entre 0 y 1 con `Math.random()`, así:

```
double randomNumber = Math.random();
```

Puedes convertir un número *decimal* a un *entero* (truncándolo), así:

```
int integerValue = (int) Math.floor(randomNumber);
```


(También hay otras formas)

(Puedes probar a [buscar en Internet](#))

## Por último

Una vez verifiques que el *test* funciona correctamente, la tarea ha sido completada.

Haz `commit` y `push` para subir los cambios al repositorio.

- 
1. ¿Crees que esta forma de gestionar los casos de error es adecuada? ¿Se te ocurre alguna forma en la que pueda dar problemas? ¿Cómo lo solucionarías? 
- 



Rubén Montero @ruben.montero changed milestone to [%Sprint 1](#) 2 weeks ago



Ania Blanco @ania.blanco mentioned in commit [47d89bca](#) 2 weeks ago