El concepto de Java constructor reference es relativamente habitual cuando trabajamos con lambdas y Streams. Sin embargo muchas veces cuesta entender como funciona y que es lo que aporta. Vamos a ver un ejemplo sencillo en el cual hacemos una referencia a un constructor con Java 8 . Para ello nos vamos a apoyar en la clase Persona.

```
package com.arquitecturajava;

public class Persona {

	private String nombre;

	public String getNombre() {
		return nombre;
	}

	public void setNombre(String nombre) {
		this.nombre = nombre;
	}

	public Persona(String nombre) {
		super();
		this.nombre = nombre;
	}


}
```

Nosotros podemos construir una lista de personas a través de un array de cadenas usando

una java constructor reference. El siguiente código nos muestra como hacerlo.

```
package com.arquitecturajava;


import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class Principal {

        public static void main(String[] args) {
        List<String> lista=Arrays.asList("pepe","juan","pedro");
        List<Persona>
listaPersonas=lista.stream().map(Persona::new).collect(Collectors.toLi
st());

        for(Persona p:listaPersonas) {
                System.out.println(p.getNombre());
        }
        }

}
```

Acabamos de crear tres personas e imprimirlas en la consola.

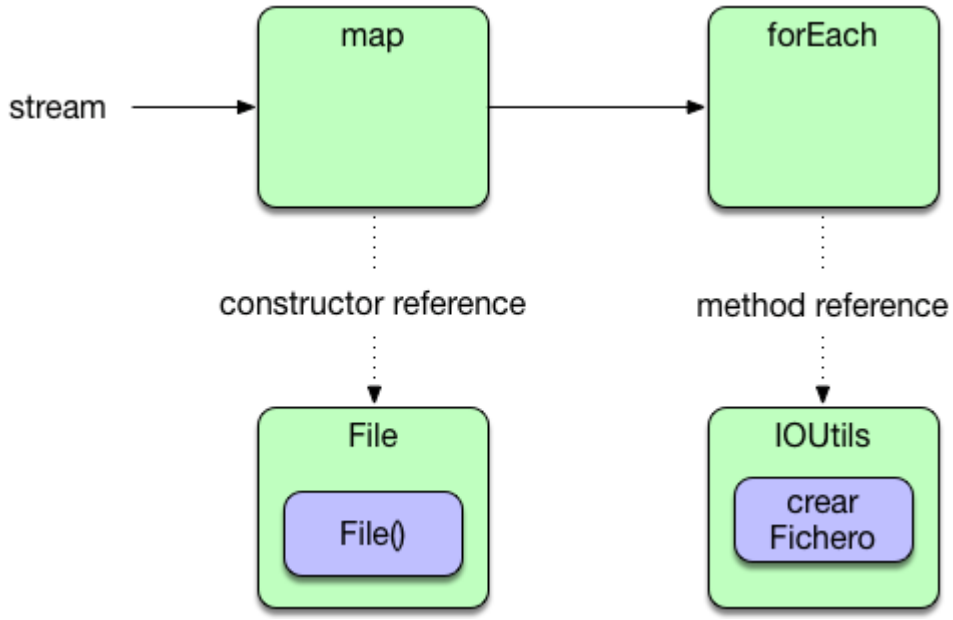
Java Constructor reference

Sin embargo es difícil ver cuando usar este tipo de sintaxis Persona:new para hacer referencia a los constructores. Un ejemplo que ayuda a clarificar es usar un array similar para construir un grupo de ficheros apoyandononos en el constructor de la clase File. Para ello nos vamos a apoyar en una clase helper que dispone de un método para crear un fichero.

```
package com.arquitecturajava;

import java.io.File;
import java.io.IOException;

public class IOUtil {

public static void  crearFicheros(File fichero) {
                try {
                        fichero.createNewFile();
                } catch (IOException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                }
        }
}
```

Podemos utilizar este clase de utilidades y su método crear fichero para crear un grupo de

ficheros usando streams y lambdas .

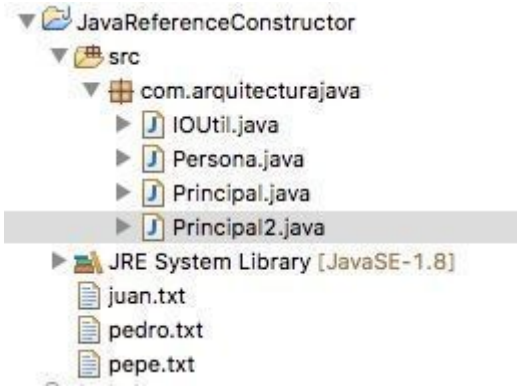


Veamos el código:

```
package com.arquitecturajava;


import java.io.File;
import java.util.Arrays;
import java.util.List;

public class Principal2 {

        public static void main(String[] args) {
        List<String>
```

```
lista=Arrays.asList("pepe.txt","juan.txt","pedro.txt");
        lista.stream().map(File::new).forEach(IOUtil::crearFicheros);


        }
}
```

Con este sencillo código a nivel de expresiones lambda y streams se crea una lista de ficheros en la carpeta de nuestro proyecto de forma rápida.



Otros artículos relacionados

1. Utilizando un Java reference method
2. Java Stream String y Java 8
3. Java Stream Collectors y su uso
4. Java Method Reference