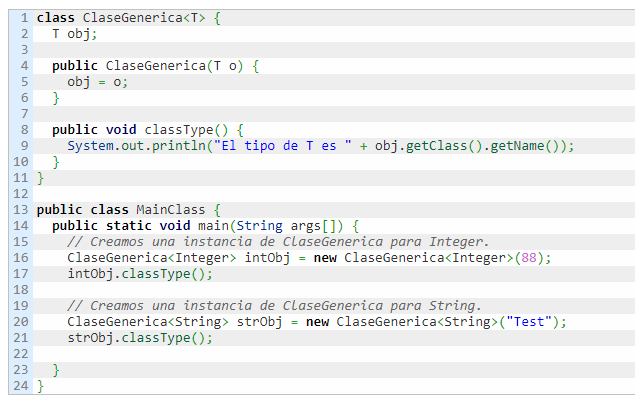
**DATOS GENÉRICOS**

En [Java](http://jonsegador.com/tag/java/), cuando definimos una nueva clase,**debemos conocer el tipo de dato**con el que trabajaremos. Si queremos realizar una operación específica dentro de esta nueva clase**, sea cual sea el tipo de datos** que va a recibir, podemos hacer uso de los **tipos genéricos**. Este tipo genérico asumirá el tipo de dato que realmente le pasaremos a la clase.

Los genéricos nos permitirán especificar qué clase de elementos han de estar dentro de una colección. Java no permitirá añadir instancias de otra clase que no sea la declarada mediante el genérico.

Reconoceremos el uso de genéricos en una clase, interfaz o método cuando veamos un solo carácter en mayúsculas encerrado en el operador “diamond”.

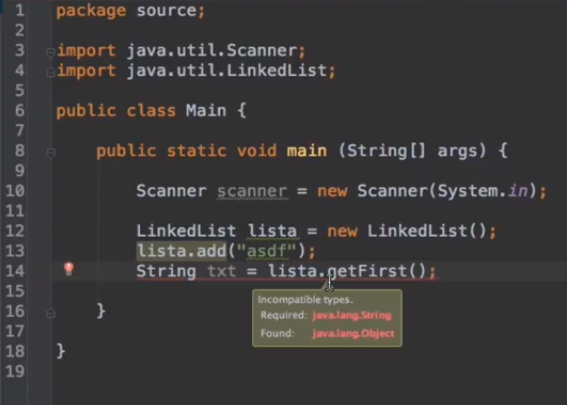
Ejemplo:

El resultado será el siguiente:

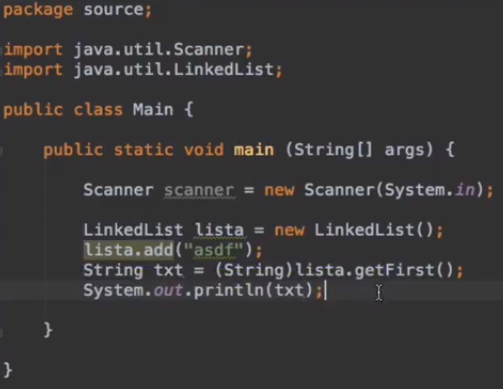
Hay que tener en cuenta que **los generics de java solo funcionan con objetos**. El código siguiente nos mostrará un error:

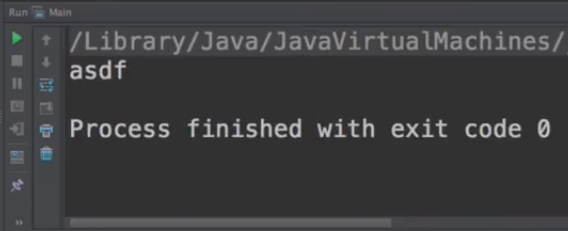
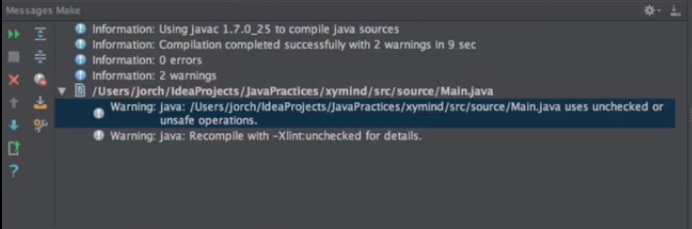


EJEMPLO 01:

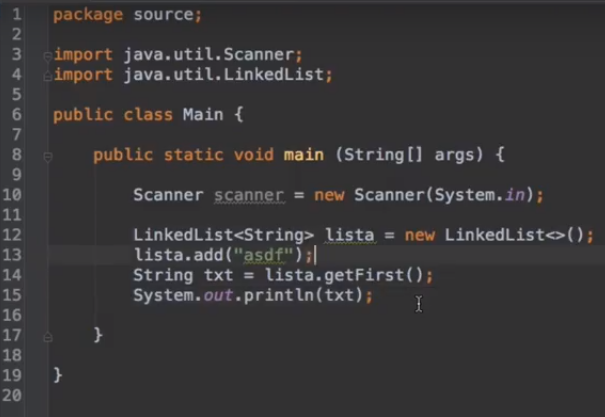


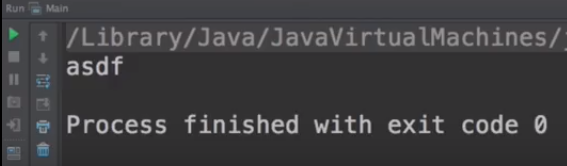




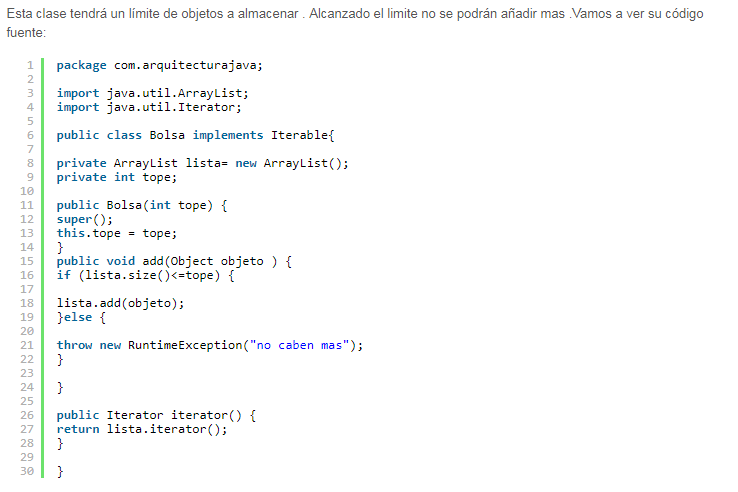
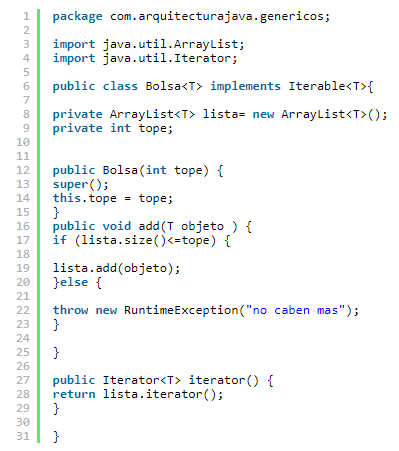
El programa funciona, pero tiene errores…

APLICAMOS GENÉRICOS:





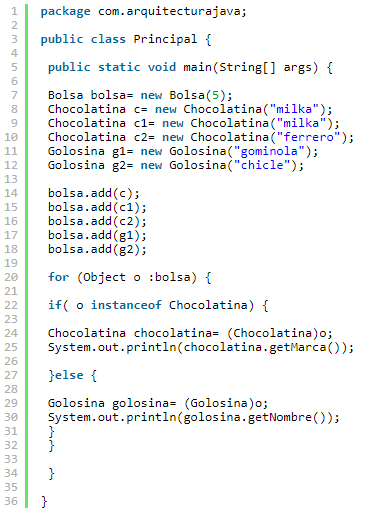
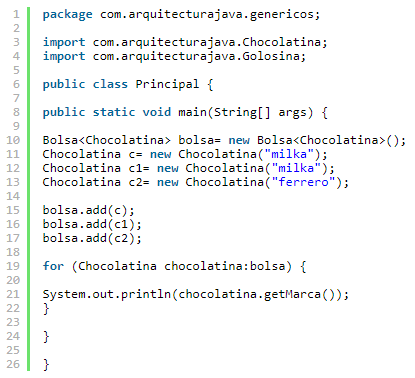
EJEMPLO 02:



El programa funcionará correctamente, pero nos podremos dar cuenta que **resulta bastante poco amigable la estructura if /else en la cual se chequean cada uno de los tipos a la hora de presentarlo por pantalla.**

**Java Generics**

Para solventar este problema podemos construir una clase Genérica. Este tipo de clase nos permitirá definir una Bolsa de un tipo concreto. Puede ser una bolsa de Golosinas o una bolsa de Chocolatinas,**pero NO de las dos cosas a la vez**. Esto en un principio puede parecer poco flexible, pero si nos ponemos a pensar cuando programamos solemos imprimir una lista de Facturas o una lista de Compras no una lista mixta.



Disponible en:

<http://jonsegador.com/2012/10/clases-y-tipos-genericos-en-java/>

<https://www.luaces-novo.es/genericos-para-recien-llegados-a-java/>