

2015

# SEXTO ENSAYO

Universidad Nacional de Ingeniería

Chunqui Vela José Lendel 20131032H



## INTERFACES

Una interface es una clase en la que todos sus métodos son abstractos, es decir, una interface es una clase abstracta pura.

Las interfaces se utilizan para establecer un “protocolo” entre clases, estableciendo una forma común para clases (nombre de métodos y sus tipos de retorno comunes).

### Utilidad de las interfaces

Lo que hacen las interfaces es separar la especificación de una clase (qué hace) de la implementación (cómo lo hace), lo que da lugar a la existencia de variables polimórficas y la invocación polimórfica de métodos.

### Forma de uso

Una interfaz se ‘implementa’ (es decir, se escriben las líneas de código correspondientes a sus métodos) en una clase mediante la palabra reservada *implements*.

El compilador se encarga que una clase que implementa una interface, declare e implemente todos los métodos de esta interface.

Declaración de una interface

```
interface nombre_interface {  
    tipo_retorno nombre_metodo ( lista_argumentos ) ;  
    ...  
}
```

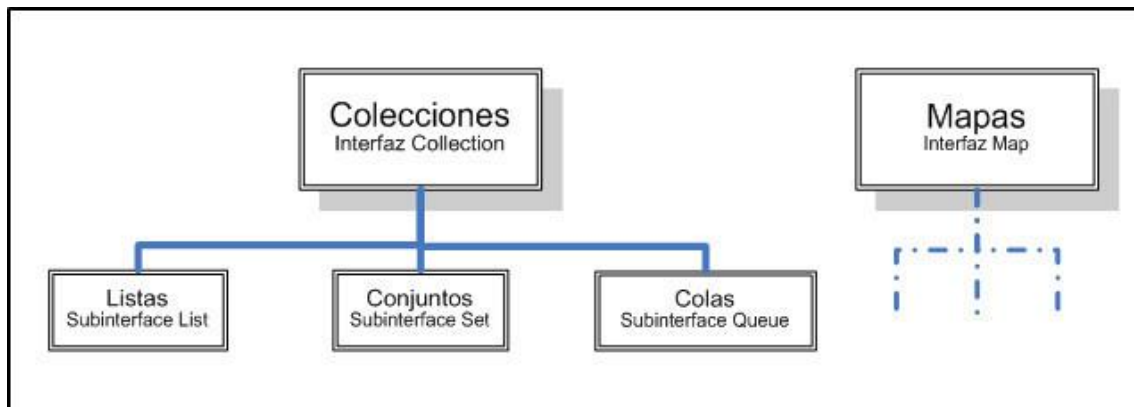
Ejemplo:

```
interface InstrumentoMusical {  
    void tocar();  
    void afinar();  
    String tipoInstrumento();  
}
```

Implementación de una interface:

```
class InstrumentoViento implements InstrumentoMusical {  
    void tocar() { . . . };  
    void afinar() { . . . };  
    String tipoInstrumento() { . . . }  
}  
  
class Guitarra extends InstrumentoViento {  
    String tipoInstrumento() {  
        return "Guitarra";  
    }  
}
```

## La interfaz Collection



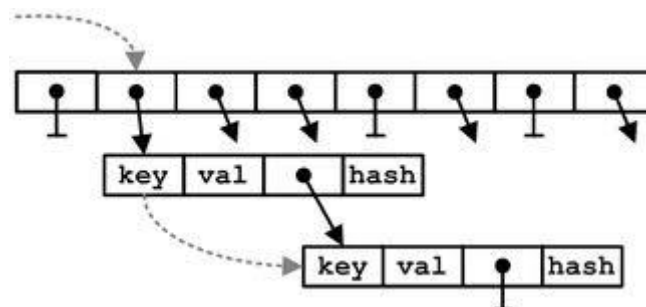
## Colección de objetos en Java:

Una colección de objetos es un objeto que puede almacenar un número variable de elementos, siendo cada elemento otro objeto.

- **Una lista (list):** Es una colección de objetos donde cada uno de ellos lleva un índice asociado.
- **Un conjunto (set):** Es una colección de objetos que no admite duplicados.
- **Una cola (queue):** Es una colección de objetos que se comportan como lo haría un grupo de personas en la cola de una caja de un supermercado. Los objetos se van poniendo en cola y el primero en salir es el primero que llegó.
- **Una pila (stack):** Es una colección de objetos que se comportan como lo haría un montón de objetos apilados, el último en añadirse queda en la cima de la pila y el primero en salir es el último que ha llegado. El primero en llegar está en el fondo de la pila.

## La interface Map

La Interface Map (java.io.Map) en Java, nos permite representar una estructura de datos para almacenar pares "clave/valor"; de tal manera que para una clave solamente tenemos un valor. Esta estructura de datos también es conocida en otros lenguajes de programación como "Diccionarios".



## PATRONES DE DISEÑO

Un patrón de diseño en informática es el patrón de las soluciones a problemas comunes en el desarrollo de Software. Es decir, un patrón de diseño nos proporciona una solución ya realizada y de comprobada efectividad a problemas de desarrollo de software en contextos similares al que estamos inmersos.

Un patrón de diseño es un conjunto de objetos y clases organizados de cierta forma que ayudan a resolver un problema recurrente en el diseño de un software de una forma limpia y reusable.

### Clases de patrones de diseño

- Patrones Creacionales: Indican la inicialización y configuración de objetos. Procuran independizar al sistema de cómo sus objetos son creados y/o representados.
- Patrones Estructurales: Separan la interfaz de la implementación. Se ocupan de cómo las clases y objetos se agrupan, para formar estructuras más grandes.
- Patrones de Comportamiento: Más que describir objetos o clases, describen la comunicación entre ellos. Se centran en los algoritmos y en la asignación de responsabilidades entre los objetos.

### El patrón Abstract Factory

Es un patrón creacional que nos permite crear, mediante una interfaz, conjuntos o familias de objetos (denominados productos) que dependen mutuamente y todo esto sin especificar cuál es el objeto concreto.

#### **Usos:**

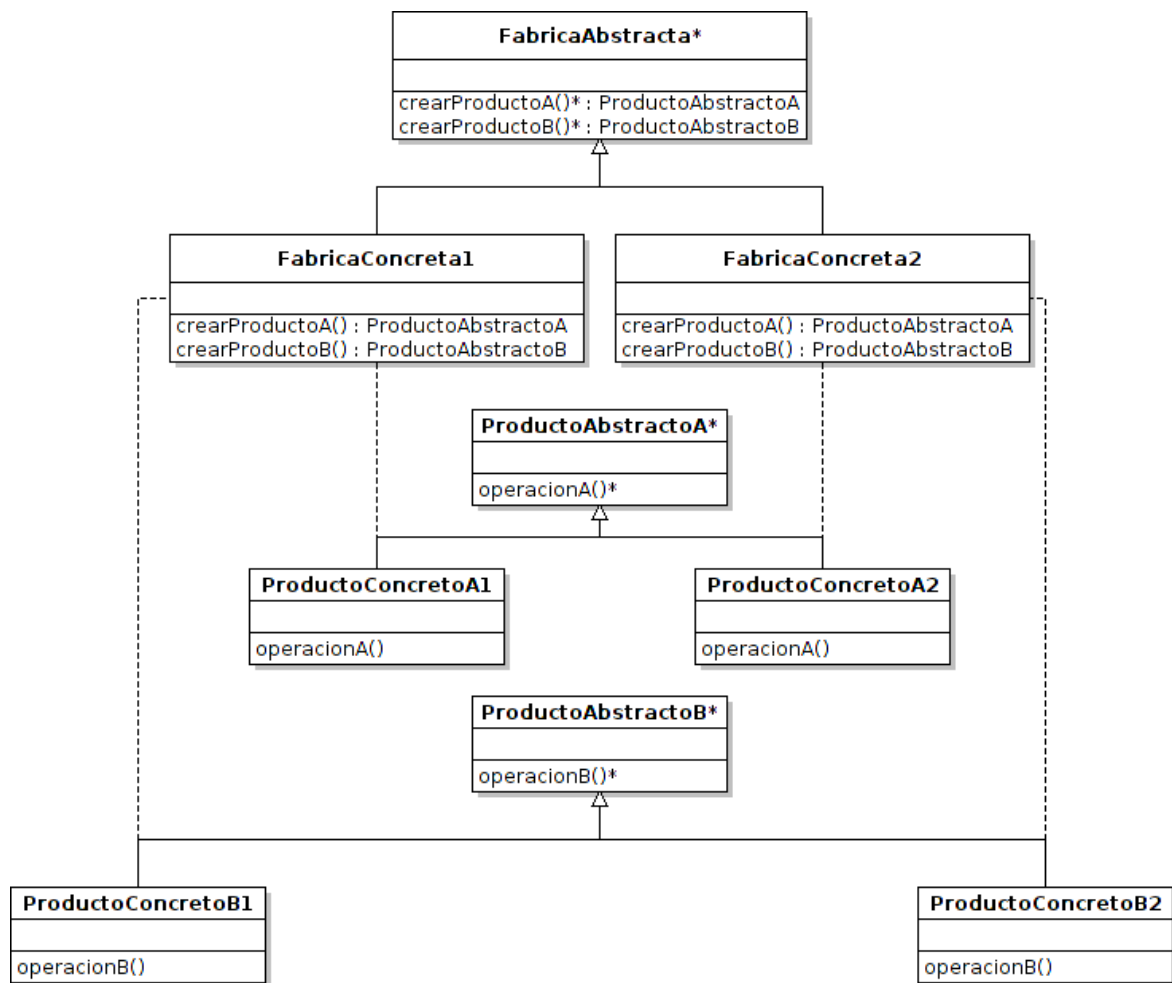
Se puede emplear este patrón de diseño cuando:

- Un sistema debe ser independiente de cómo sus objetos son creados.
- Un sistema debe ser 'configurado' con una cierta familia de productos.
- Se necesita reforzar la noción de dependencia mutua entre ciertos objetos.

### **Elementos del Patrón Abstract Factory**

- **Fabrica Abstracta:** Define un conjunto de métodos (interfaz) para la creación de productos abstractos.
- **Fabrica Concreta 1/2:** Implementa la interfaz de la Fabrica Abstracta para la creación de los distintos productos concretos.
- **Producto Abstracto A/B:** Define la interfaz de los objetos de tipo Producto A/B.
- **Producto Concreto A1/A2/B1/B2:** Implementan su respectiva interfaz representando un producto concreto.

## Estructura del Patrón Abstract Factory



## Bibliografía:

### Interfaces

<http://www.arrakis.es/~abelp/ApuntesJava/Interfaces.htm>

[http://aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=671:para-que-sirven-las-interfaces-java-implementar-una-interfaz-del-api-ventajas-y-ejemplos-basicos-cu00697b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188](http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=671:para-que-sirven-las-interfaces-java-implementar-una-interfaz-del-api-ventajas-y-ejemplos-basicos-cu00697b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188)

[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=630:coleccion-de-objetos-de-tamano-variable-java-listas-pilas-colas-collection-list-set-queue-map-cu00664b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188](http://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=630:coleccion-de-objetos-de-tamano-variable-java-listas-pilas-colas-collection-list-set-queue-map-cu00664b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188)

### Patrones de diseño

<https://msdn.microsoft.com/es-es/library/bb972240.aspx>

[http://es.wikipedia.org/wiki/Patr%C3%B3n\\_de\\_dise%C3%B1o](http://es.wikipedia.org/wiki/Patr%C3%B3n_de_dise%C3%B1o)

<http://lineadecodigo.com/patrones/patron-abstract-factory/>