



IES SALADILLO – CFGS DAM
PROGRAMACIÓN
Curso 2015-2016
Herencia
Fecha: 07/03/2016

Apellidos y Nombre: _____

Nota:

- El programa, además de funcionar, tiene que ser elaborado aplicando de manera óptima la teoría de herencia de java.

Examen: Realiza un diagrama de clases y un programa en Java para realizar este ejercicio:

La directora del centro nos ha pedido que le hagamos una aplicación para el instituto. Necesitamos registrar a todos los alumnos y profesores. De ambos queremos saber su nombre. De los alumnos necesitamos además un array de notas y de los profesores el sueldo. Ambos contendrán el método **tipo_numero** que devolverá la siguiente cadena: "Soy el alumno número x" para los alumnos y "Soy el profesor número x" para los profesores donde x será el orden de creación en la aplicación. **Modelar usando herencia entre clases.**

La directora nos ha insistido en que por favor no permitamos registrar a nadie cuyo nombre no cumpla con los siguientes requisitos:

- Solamente contenga letras y espacios.
- Contenga el nombre y los dos apellidos. Tanto el nombre como los apellidos deben empezar en mayúsculas y seguirles minúsculas. Si alguno de ellos es compuesto, la segunda parte debe empezar en minúscula. Ejemplo: Jose manuel Contreras De los ríos
- El nombre y los dos apellidos deben ir en el atributo nombre.

Si un nombre no cumple estos requisitos no se debe permitir:

- Crear el objeto.
- Actualizar el atributo nombre de un objeto ya creado.

Para ello, crear la excepción **NombreIncorrectoException**.

La directora quiere validar personalmente todas las personas para dejar constancia de que efectivamente ese alumno está matriculado y de que ese profesor trabaja en el centro. Para ello, vamos a tener un método validar. Además, necesitamos un método para ponerle una nota al alumno y otro método para aumentarle el sueldo al profesor. **Modelar usando herencia entre interfaces.**

Realiza el equals, clone y toString.

Realizar una **aplicación** en una clase llamada lesSaladillo que haga lo siguiente:

1. Intentar crear un alumno con un nombre no correcto. No lo debe permitir. Mostrar un mensaje utilizando una variable de excepción.
2. Crear 3 alumnos y 3 profesores y meterlos en un array.
3. Ejecutar de cada persona del array el método tipo_numero y a continuación el toString.
4. Crear otro array para introducir los clones.
5. Utiliza equals para comprobar que los clones son iguales a sus originales.
6. Realizar un **método** que tenga como parámetro un array de variables de interfaz. El método recorrerá el array y hará lo siguiente: (todo el código se tiene que realizar a nivel de interfaces)
 - ✓ Validará las personas de las posiciones pares.
 - ✓ Si es un alumno, le añadirá una nota nueva.
 - ✓ Si es un profesor, le aumentará el sueldo un 10%.

La aplicación llamará a este método con el array de los clonados. Luego, utiliza equals para comprobar que los clones ya no son iguales a sus originales.

Firma del alumno