

Trabajo Tema 2

- Se propone un pequeño reto tecnológico consistente en **averiguar**, mediante los mecanismos de **reflexión** y **serialización** del lenguaje Java, los atributos y métodos de las instancias pertenecientes una clase que, en adelante, referiremos como “**clase a descubrir**” y cuyo bytecode está disponible en alguna **dirección de Internet**
- Dicha dirección de Internet puede obtenerse invocando el método `public String obtenerRutaCompleta()` de una instancia serializada de la clase `ClaseASerializar` almacenada en el fichero “**fichero.salida**” disponible en la zona de descargas de **Tutor**. Allí mismo también se encuentra disponible el fichero “**ClaseASerializar.class**” con el bytecode de la mencionada clase, necesario para deserializar el objeto de “**fichero.salida**”
- Se debe proporcionar una **implementación** que, tomando como **entrada** el **DNI/NIE** sin letras del **alumno** y la **ruta** hasta “**fichero.salida**” desde el directorio de trabajo actual, muestre el resultado de la ejecución del método `public int computa (String dni)` sobre una instancia de la “**clase a descubrir**”
- El trabajo tiene carácter individual. Se valorará la cantidad de información proporcionada acerca de la “**clase a descubrir**”
- Debe entregarse un fichero .java, así como una captura de pantalla con el resultado de la ejecución del programa implementado
- **El plazo de entrega finaliza el 7-jun-2016 a las 12h**

Trabajo Tema 2

- A continuación se proporcionan algunas funciones que pueden resultar útiles para la realización de este trabajo:

En ClaseASerializar

- `public String obtenerNombreClase()` //Devuelve el nombre de la **clase a descubrir**
- `public String obtenerDirecciónClase()` //Devuelve el directorio donde se encuentra el bytecode de la **clase a descubrir**
- `public String obtenerRutaCompleta()` //Devuelve `obtenerDirecciónClase()+obtenerNombreClase()+".class";`

En general:

- Clase `URLClassLoader`
- `Thread.currentThread().setContextClassLoader(ClassLoader)`
- `unaCadena.split(".class")[0]` // Para obtener el nombre de una clase sin “.class”

`java.lang.Class`

- `public Method getMethod(String name, Class<?>... parameterTypes)`

`java.lang.reflect.Method`

- `public Object invoke(Object obj, Object... args)`