



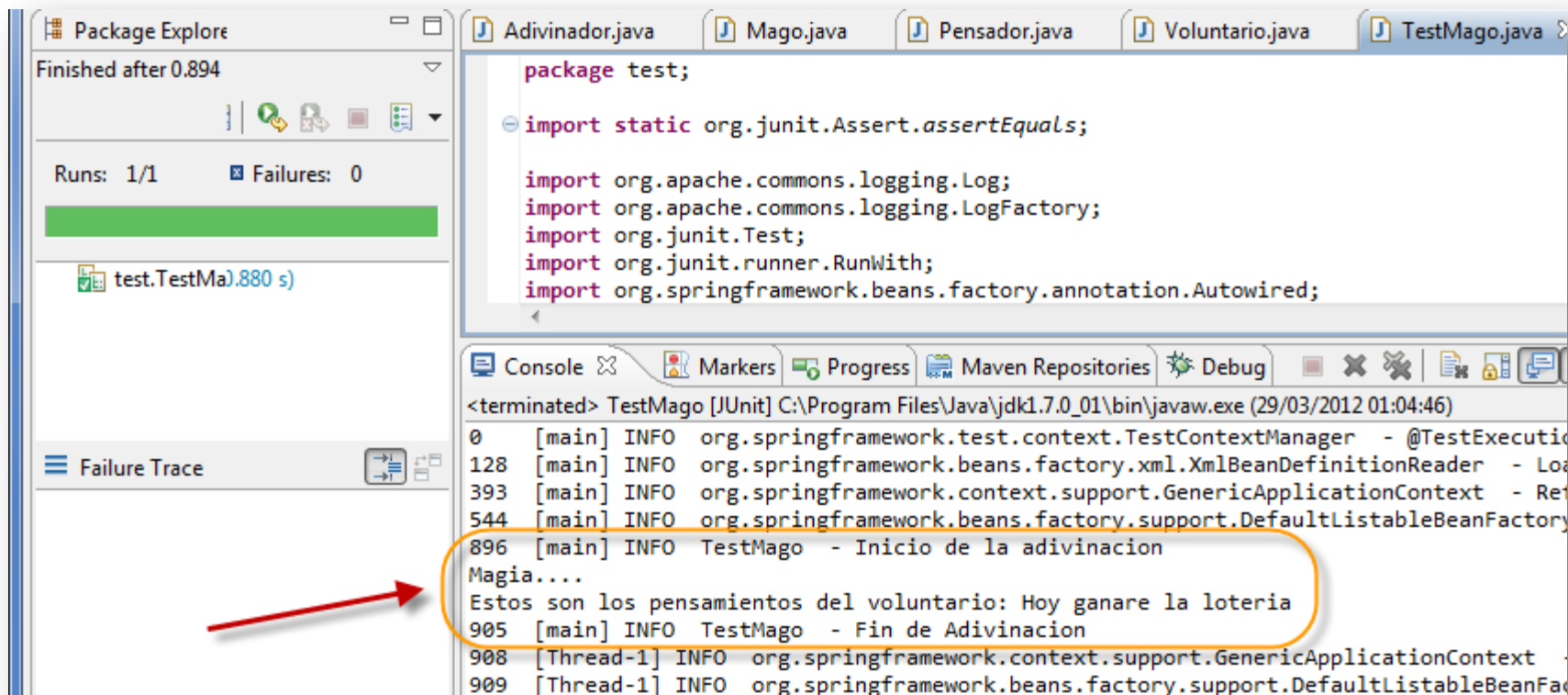
Ejercicio

Proyecto Concurso de Talentos v8

Objetivo del Ejercicio

- El objetivo del ejercicio es agregar funcionalidad AOP con el uso de Anotaciones y el paso de parámetros.

Al finalizar deberemos ver la siguiente salida, resultado de aplicar los conceptos de AOP descritos:



The screenshot shows an IDE with the following components:

- Package Explorer:** Shows the test results for `test.TestMago` with a green bar indicating success. It states "Finished after 0.894" and "Runs: 1/1 Failures: 0".
- Source Editor:** Displays the `TestMago.java` file with the following code:

```
package test;

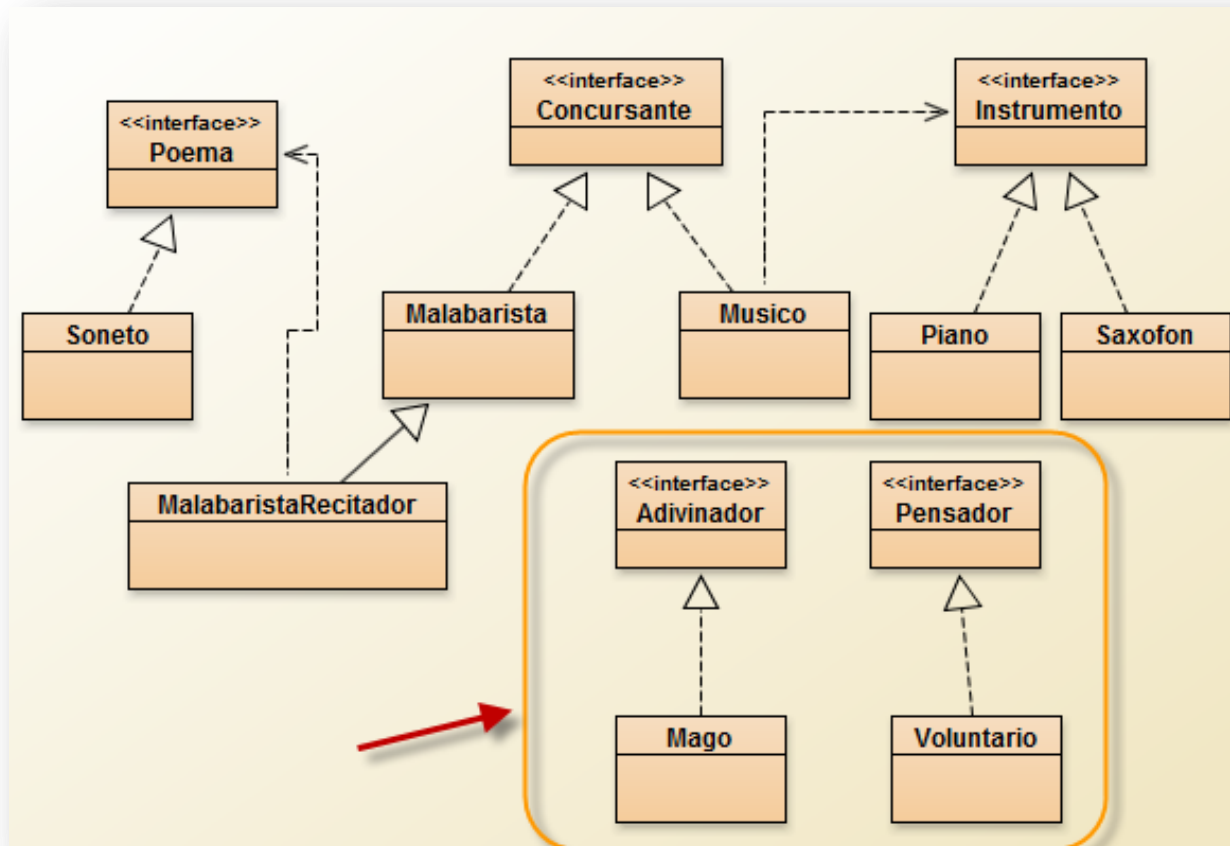
import static org.junit.Assert.assertEquals;

import org.apache.commons.logging.Log;
import org.apache.commons.logging.LogFactory;
import org.junit.Test;
import org.junit.runner.RunWith;
import org.springframework.beans.factory.annotation.Autowired;
```
- Console:** Shows the output of the test execution. A red arrow points to the following log messages:

```
<terminated> TestMago [JUnit] C:\Program Files\Java\jdk1.7.0_01\bin\javaw.exe (29/03/2012 01:04:46)
0 [main] INFO org.springframework.test.context.TestContextManager - @TestExecution
128 [main] INFO org.springframework.beans.factory.xml.XmlBeanDefinitionReader - Loa
393 [main] INFO org.springframework.context.support.GenericApplicationContext - Ret
544 [main] INFO org.springframework.beans.factory.support.DefaultListableBeanFactory
896 [main] INFO TestMago - Inicio de la adivinacion
Magia....
Estos son los pensamientos del voluntario: Hoy ganare la loteria
905 [main] INFO TestMago - Fin de Adivinacion
908 [Thread-1] INFO org.springframework.context.support.GenericApplicationContext
909 [Thread-1] INFO org.springframework.beans.factory.support.DefaultListableBeanFac
```

Diagrama de Clases

Al finalizar deberemos tener el Proyecto Concurso Talentos con las siguientes clases:

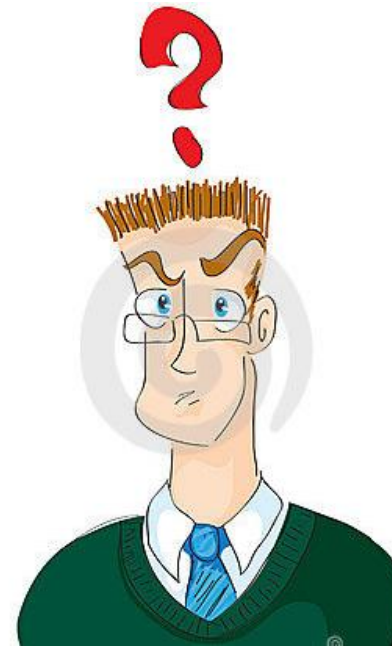


Mago Adivinador de Pensamientos

- Demos la bienvenida a un nuevo concursante, el cual es un mago que puede adivinar los pensamientos (telepatía).
- Así que a continuación agregaremos algunas clases más a nuestro proyecto para definir las características de este virtuoso mago.



Mago Adivinador de Pensamientos



Voluntario (Pensador)



Paso 1. Crear la Interface Adivinador

Debemos crear la Interfaz Adivinador.java y agregamos el siguiente código:

```
package concursantes;

public interface Adivinador {

    public void interceptarPensamientos(String pensamientos);

    public String getPensamientos();
}
```

Paso 2. Agregar la clase Mago

Crear la clase Mago.java y agregar el siguiente código:

```
package concursantes;

import org.aspectj.lang.annotation.Aspect;
import org.aspectj.lang.annotation.Before;
import org.aspectj.lang.annotation.Pointcut;
import org.springframework.stereotype.Component;

@Component
@Aspect
public class Mago implements Adivinador {

    private String pensamientos;

    @Pointcut("execution(* concursantes.Pensador.pensarEnAlgo(String)) && args(pensamientos)")
    public void pensar(String pensamientos) {
    }

    @Before("pensar(pensamientos)")
    public void interceptarPensamientos(String pensamientos) {
        System.out.println("Magia....");
        System.out.println("Estos son los pensamientos del voluntario: " + pensamientos);
        this.pensamientos = pensamientos;
    }

    public String getPensamientos() {
        return this.pensamientos;
    }
}
```



Paso 3. Crear la Interface Pensador

Debemos crear la Interfaz Pensador.java y agregamos el siguiente código:

```
package concursantes;

public interface Pensador {

    void pensarEnAlgo(String pensamientos);
}
```



Paso 4. Agregar la clase Voluntario

Crear la clase Voluntario.java y agregar el siguiente código:

```
package concursantes;

import org.springframework.stereotype.Component;

@Component
public class Voluntario implements Pensador {

    private String pensamientos;

    public void pensarEnAlgo(String pensamientos) {
        this.pensamientos = pensamientos;
    }

    public String getPensamientos() {
        return this.pensamientos;
    }
}
```


Paso 5. Crear la clase TestMago.java

Crear la clase JUnit llamada TestMago.java y agregar el siguiente código:

```
package test;

import static org.junit.Assert.assertEquals;

import org.apache.commons.logging.Log;
import org.apache.commons.logging.LogFactory;
import org.junit.Test;
import org.junit.runner.RunWith;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.test.context.ContextConfiguration;
import org.springframework.test.context.junit4.SpringJUnit4ClassRunner;

import concursantes.Adivinador;
import concursantes.Pensador;

@RunWith(SpringJUnit4ClassRunner.class)
@ContextConfiguration(locations = {"classpath:applicationContext.xml"})
public class TestMago {

    private static Log logger = LogFactory.getLog("TestMago");
    @Autowired
    private Pensador voluntario;
    @Autowired
    private Adivinador mago;

    @Test
    public void testMagoAdivinador() {
        logger.info("Inicio de la adivinacion");

        voluntario.pensarEnAlgo("Hoy ganare la loteria");

        assertEquals("Hoy ganare la loteria", mago.getPensamientos());

        logger.info("Fin de Adivinacion");
    }
}
```

Paso 6: Ejecutar la Prueba

- Ejecutamos la clase TestMago.java como una prueba de Junit y observamos el siguiente resultado:

