

```
/*
Pro Spring
By Rob Harrop
Jan Machacek
ISBN: 1-59059-461-4
Publisher: Apress
*/
```

```
//File: beans.xml
```

```
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN" "http://www.springframework.org/dtd/spring-
beans.dtd">
```

```
<beans>
  <bean id="provider" class="HelloWorldModel">
    <constructor-arg>
      <value>This is a configurable message</value>
    </constructor-arg>
  </bean>

  <bean id="renderer" class="StandardOutView">
    <property name="model">
      <ref local="provider"/>
    </property>
  </bean>
</beans>
```

```
////////////////////////////////////
```

```
public interface Model {

  public String getMessage();
}
```

```
////////////////////////////////////
```

```
public interface View {

  public void render();

  public void setModel(Model m);
  public Model getModel();
}
```

```
////////////////////////////////////
```

```
public class StandardOutView implements View {

  private Model model = null;

  public void render() {
    if (model == null) {
      throw new RuntimeException(
        "You must set the property model of class:"
      );
    }
  }
}
```

```

        + StandardOutView.class.getName());
    }

    System.out.println(model.getMessage());
}

public void setModel(Model m) {
    this.model = m;
}

public Model getModel() {
    return this.model;
}
}
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

```

```

public class HelloWorldModel implements Model {
    String mess;

    public HelloWorldModel(String m){
        mess = m;
    }

    public String getMessage() {

        return mess;
    }
}
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

```

```

import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.FileSystemResource;

public class HelloWorldXmlWithDI {

    public static void main(String[] args) throws Exception {

        // get the bean factory
        BeanFactory factory = getBeanFactory();
        View mr = (View) factory.getBean("renderer");
        mr.render();
    }

    private static BeanFactory getBeanFactory() throws Exception {
        // get the bean factory
        BeanFactory factory = new XmlBeanFactory(new FileSystemResource(
            "build/beans.xml"));

        return factory;
    }
}

```

