XML Configuration

XML Configuration

```
<beans  xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="
      http://www.springframework.org/schema/beans
      http://www.springframework.org/schema/beans/spring-beans.xsd">
</beans>
```

- **xmlns="http://www.springframework.org/schema/beans"** defines a default namespace for tags
- **xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"** standard namespace used to validate that an XML file is following the schema appropriately
- **xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">** used by the validator parser to validate if an XML is according to the schema (every tag in *beans* namespace validate against *spring-beans.xsd*)

XML Configuration

XML	Java

`<bean id="dataSource" class="org.springframework.jdbc.datasource.SimpleDriverDataSource"/>`	`@Bean`
	`public DataSource dataSource() {`
	`  return new SimpleDriverDataSource();`
	`}`

`<context:component-scan base-package="rs.saga.businessobject"/>`	`@ComponentScan(basePackages = {"rs.saga.businessobject"})`

Injecting Non Beans Dependency

```
<bean id="player" class="rs.saga.businessobject.Player">
   <constructor-arg name="age" value="20"/>
   <constructor-arg name="firstName" value="John"/>
   <constructor-arg name="lastName" value="Doe"/>
</bean>
```

- The Spring container knows how to convert all primitive types and their reference wrapper types. String values, booleans, and numeric types are supported by default.

```
<constructor-arg name="dateOfBirth" value="1998/10/10"/>
```

- Spring container can convert any the value of the attribute to the type using property editor.

```
   <bean id="numberFormat" class="java.text.NumberFormat" factory-method="getCurrencyInstance"/>
```

- Bean element support creation using Factory Beans

XML Configuration

- Advantages:
  - For some libraries the only configuration option
  - Namespaces offer shortcuts that are simpler than annotation or Java config

- Disadvantages:
  - Tedious
  - No compile time check
  - Complex when it comes to factory objects