**Important note: Before going further you are requested to go through theory of 'how to deal with collections in Spring Framework' mentioned in JavaLive Notes on Spring Core thoroughly so that you can understand the concept very well. (\*\*This activity i.e. going through theory part of respective subject is applicable for each of our project.)**

**What this project does??**

This project explains how to deal with collections in spring.

**Note that this is xml based spring project having configuration file viz. ' spring-config.xml' in resources folder.**

**Steps to create project:-**

1. Create quick start maven project. Refer file 'CreateQuickStartMavenProjectInEclipse.docx' in 'SprinDIWithAutowiring' project for more details.

2. Add require dependencies for spring as shown in pom.xml file.

3. Create required packages and add the files.

**Functioning of the project**

This project is mainly divided in three parts:

1. com.javaLive.beans.list, com.javaLive.beans.map, com.javaLive.beans.properties packages contains respective classes. We are going to use these classes in spring-config file.

2. spring-config.xml file in resources folder contains bean definition for various collections.

3. Classes in com.javaLive.main package contains respective classes to utilize declared collection.

**Very important mechanism of Spring is as follows:**

1. for example com.javaLive.properties.DataSource contains class definition only. It does not does not have any initialization.

2. Bean configuration file contains bean declaration scope and initialization and reference values.

3. Bean utilization class (PropertiesMainClass.java in this case) is get that class from Spring ApplicationContext container as that is responsible for Beans lifecycle now and use it in the program.

So this is we can say a 'Modus Operandy' of Spring Framework.

And Spring's configuration file is driving force of the application.