         <properties>

                 <spring.version>4.3.5.RELEASE</spring.version>

                 <maven.compiler.target>1.8</maven.compiler.target>

                 <maven.compiler.source>1.8</maven.compiler.source>

         </properties>

**Step-1**

  <!-- JAX-WS This dependency for METRO framework which is used to develop

                          soap based web service -->

                 <dependency>

                          <groupId>com.sun.xml.ws</groupId>

                          <artifactId>jaxws-rt</artifactId>

                          <version>2.2.8</version>

                 </dependency>

         <!-- <https://mvnrepository.com/artifact/org.springframework/spring-core> -->

                 <dependency>

                          <groupId>org.springframework</groupId>

                          <artifactId>spring-core</artifactId>

                          <version>${spring.version}</version>

                 </dependency>

                 <dependency>

                          <groupId>org.springframework</groupId>

                          <artifactId>spring-context</artifactId>

                          <version>${spring.version}</version>

                 </dependency>

                 <!-- Spring web module -->

                 <dependency>

                          <groupId>org.springframework</groupId>

                          <artifactId>spring-web</artifactId>

                          <version>${spring.version}</version>

                 </dependency>

                 <!-- This is responsible for spring and metro integration-->

                          <dependency>

                                      <groupId>org.jvnet.jax-ws-commons.spring</groupId>

                                      <artifactId>jaxws-spring</artifactId>

                                      <version>1.9</version>

                                                     <exclusions>

                                      <exclusion>

                                                <groupId>javax.servlet</groupId>

                                                <artifactId>servlet-api</artifactId>

                                      </exclusion>

                                      <exclusion>

                                                <groupId>org.springframework</groupId>

                                                <artifactId>spring</artifactId>

                                      </exclusion>

                                      <exclusion>

                                                <groupId>org.springframework</groupId>

                                                <artifactId>spring-core</artifactId>

                                      </exclusion>

                                      <exclusion>

                                                <groupId>org.springframework</groupId>

                                                <artifactId>spring-context</artifactId>

                                      </exclusion>

                                       <exclusion>

                                                <groupId>org.springframework</groupId>

                                                <artifactId>spring-web</artifactId>

                                      </exclusion>

                                      <exclusion>

                                                <groupId>org.jvnet.staxex</groupId>

                                                <artifactId>stax-ex</artifactId>

                                      </exclusion>

                                    </exclusions>

                          </dependency>

**Step-2**

Go & annotated your web service provider with spring annotation

//JAX-WS

@WebService

@Service("PassportSoapEndpoint")

**public** **class** PassportSoapEndpoint {

         @Autowired

         @Qualifier("PassportServiceImpl")

**private** PassportService passportService;

**public** String uploadPassport(Passport passport) {

                 passportService.addPassport(passport);

                 System.***out***.println("passport = "+passport);

**return** "success";

         }

**public** Passport findPassportDetailByNumber(String pnumber) {

                 Passport passport=**new** Passport();

                 passport.setAddress("CA1202");

                 passport.setCountry("India");

                 passport.setExpiryDate("12-July-2019");

                 passport.setFirstName("Nagendra");

                 passport.setLastName("Kumar");

                 passport.setMarritalStatus("Married");

                 passport.setNumber("GA322312902");

**return** passport;

         }

**Step-3**

**Rename your sun-jaxws.xml  ->>>>>>>  we-donot-need-this.file-tsun-jaxws.xml**

**Step-4**

Create new file for spring root web container

/WEB-INF/zara-root-web-appication-context.xml

**Step-5**

Scan the package in which your spring annotated web service endpoints are present and map your web service as we did inside sun-jaxws.xml

<beans xmlns=*"*[*http://www.springframework.org/schema/beans*](http://www.springframework.org/schema/beans)*"*

       xmlns:xsi=*"*[*http://www.w3.org/2001/XMLSchema-instance*](http://www.w3.org/2001/XMLSchema-instance)*"*

       xmlns:aop=*"*[*http://www.springframework.org/schema/aop*](http://www.springframework.org/schema/aop)*"*

       xmlns:tx=*"*[*http://www.springframework.org/schema/tx*](http://www.springframework.org/schema/tx)*"*

       xmlns:ws=*"*[*http://jax-ws.dev.java.net/spring/core*](http://jax-ws.dev.java.net/spring/core)*"*

       xmlns:context=*"*[*http://www.springframework.org/schema/context*](http://www.springframework.org/schema/context)*"*

       xmlns:wss=*"*[*http://jax-ws.dev.java.net/spring/servlet*](http://jax-ws.dev.java.net/spring/servlet)*"*

    xsi:schemaLocation=*"http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-3.2.xsd*

[*http://www.springframework.org/schema/aop*](http://www.springframework.org/schema/aop)[*http://www.springframework.org/schema/aop/spring-aop-3.2.xsd*](http://www.springframework.org/schema/aop/spring-aop-3.2.xsd)

[*http://www.springframework.org/schema/context*](http://www.springframework.org/schema/context)[*http://www.springframework.org/schema/context/spring-context-3.0.xsd*](http://www.springframework.org/schema/context/spring-context-3.0.xsd)

[*http://jax-ws.dev.java.net/spring/core*](http://jax-ws.dev.java.net/spring/core)[*http://jax-ws.dev.java.net/spring/core.xsd*](http://jax-ws.dev.java.net/spring/core.xsd)

[*http://jax-ws.dev.java.net/spring/servlet*](http://jax-ws.dev.java.net/spring/servlet)[*http://jax-ws.dev.java.net/spring/servlet.xsd*](http://jax-ws.dev.java.net/spring/servlet.xsd)

[*http://www.springframework.org/schema/tx*](http://www.springframework.org/schema/tx)[*http://www.springframework.org/schema/tx/spring-tx-3.2.xsd*](http://www.springframework.org/schema/tx/spring-tx-3.2.xsd)*"*

       default-lazy-init=*"true"*>

       <import resource=*"passport-service.xml"*/>

      <context:component-scan base-package=*"com.soap.web.service"*/>

       <wss:binding url=*"/passportSoapEndpoint"*>

        <wss:service>

            <ws:service bean=*"#PassportSoapEndpoint"*>

            </ws:service>

        </wss:service>

    </wss:binding>

       <wss:binding url=*"/factorialEndpoint"*>

        <wss:service>

            <ws:service bean=*"#FactorialEndpoint"*>

            </ws:service>

        </wss:service>

    </wss:binding>

</beans>

**Step-6**

**do following changes inside web.xml**

  <!-- This is responsible for creating

   spring root web container

    /WEB-INF/applicationContext.xml

   -->

   <!-- Below is the name of the spring configuration file

         which will read by ContextLoaderListener

    -->

   <context-param>

                 <param-name>contextConfigLocation</param-name>

                 <param-value>/WEB-INF/zara-root-web-appication-context.xml</param-value>

   </context-param>

  <listener>

                 <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>

  </listener

            <!--

         WSServlet - renamed as WSSpringServlet will forward your request to soap web service container

         against requested url-pattern weatherSoapProvider

         WSServlet is creating web service container where your all

         web services will be up and running

          -->

         <servlet>

        <servlet-name>metro</servlet-name>

        <servlet-class>

                com.sun.xml.ws.transport.http.servlet.WSSpringServlet

        </servlet-class>

        <load-on-startup>1</load-on-startup>

    </servlet>















