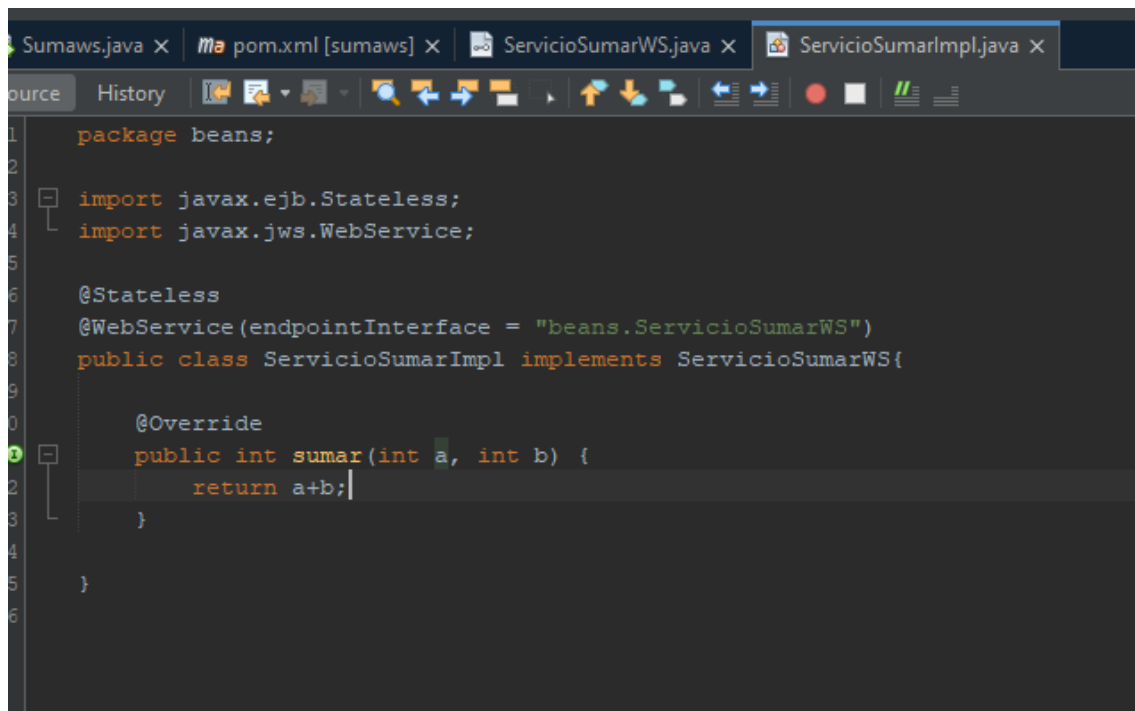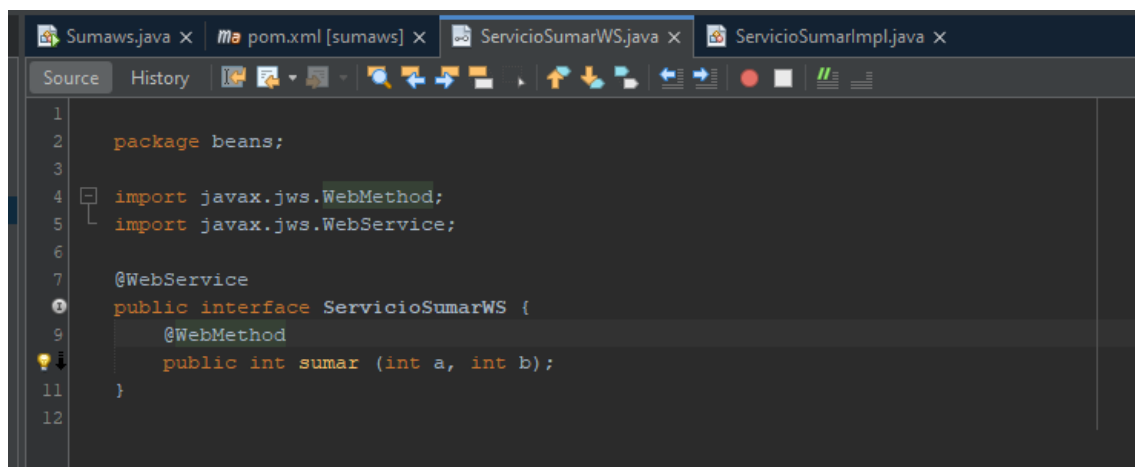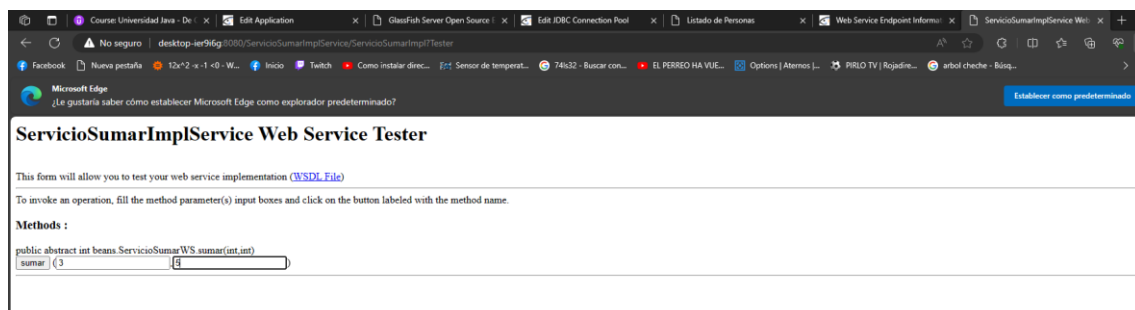# Hola mundo web services con jax-ws en Java EE 8


```java
package beans;

import javax.ejb.Stateless;
import javax.jws.WebService;

@Stateless
@WebService(endpointInterface = "beans.ServicioSumarWS")
public class ServicioSumarImpl implements ServicioSumarWS{

    @Override
    public int sumar(int a, int b) {
        return a+b;
    }

}
```



```java
package beans;

import javax.jws.WebMethod;
import javax.jws.WebService;

@WebService
public interface ServicioSumarWS {
    @WebMethod
    public int sumar (int a, int b);
}
```


ServicioSumarImplService Web Service Tester

This form will allow you to test your web service implementation (WSDL File)

To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.

Methods :

public abstract int beans.ServicioSumarWS.sumar(int,int)

sumar (3 , 5)

sumar Method invocation

Method parameter(s)

Type	Value
int	3
int	5

Method returned

int : "**8**"

SOAP Request

```xml
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <S:Body xmlns:ns2="http://beans/">
        <ns2:sumar>
            <arg0>3</arg0>
            <arg1>5</arg1>
        </ns2:sumar>
    </S:Body>
</S:Envelope>
```
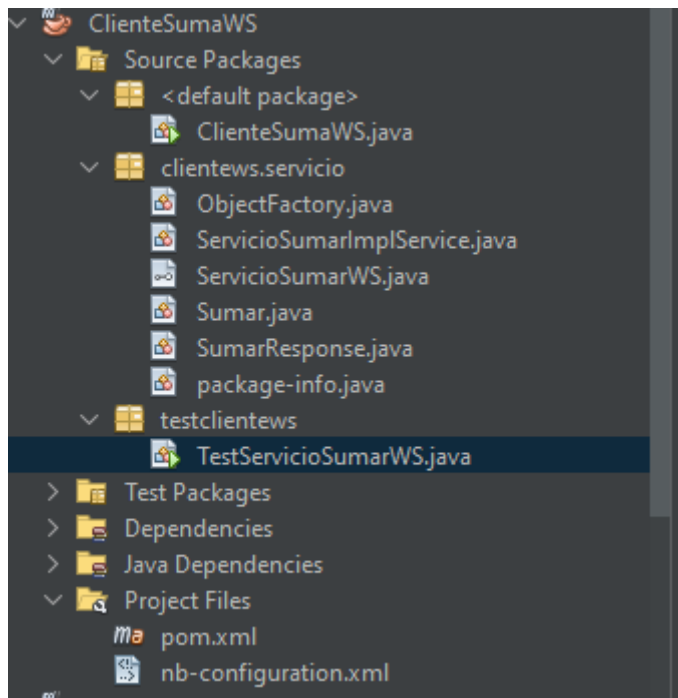

SOAP Response

```xml
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <S:Body xmlns:ns2="http://beans/">
        <ns2:sumarResponse>
            <return>8</return>
        </ns2:sumarResponse>
    </S:Body>
</S:Envelope>
```

Cliente HolaMundo Web Services con JAX-WS.


```xml
    </properties>
    <dependencies>
        <dependency>
            <groupId>javax</groupId>
            <artifactId>javaee-api</artifactId>
            <version>8.0.1</version>
            <scope>provided</scope>
        </dependency>
        <dependency>
            <groupId>javax.jws</groupId>
            <artifactId>javax.jws-api</artifactId>
            <version>1.1</version>
            <scope>provided</scope>
        </dependency>
    </dependencies>
```

```xml
<build>
    <plugins>
        <plugin>
            <groupId>org.codehaus.mojo</groupId>
            <artifactId>jaxws-maven-plugin</artifactId>
            <version>2.5</version>
            <configuration>
                <vmArgs>
                    <vmArg>-Djavax.xml.accessExternalSchema=all</vmArg>
                </vmArgs>
            </configuration>
            <executions>
                <execution>
                    <goals>
                        <goal>wsimport</goal>
                    </goals>
                    <configuration>
                        <wsdlUrls>
                            <wsdlUrl>http://desktop-ier9i6g:8080/ServicioSumarImplService/ServicioSumarImpl?wsdl</wsdlUrl>
                        </wsdlUrls>
                        <packageName>clientews.servicio</packageName>
                        <sourceDestDir>${basedir}/src/main/java</sourceDestDir>
                    </configuration>
                </execution>
            </executions>
        </plugin>
    </plugins>
</build>
```

```java
package testclientews;

import clientews.servicio.ServicioSumarImplService;
import clientews.servicio.ServicioSumarWS;

public class TestServicioSumarWS {
    public static void main(String[] args) {
        ServicioSumarWS serviciosumar= new ServicioSumarImplService().getServicioSumarImplPort();
        System.out.println(x: "Ejecutando servicio sumar ws");
        System.out.println(x: serviciosumar.sumar(arg0: 6, arg1: 3));
        System.out.println(x: "Fin Servicio sumar");
    }
}
```

testclientews.TestServicioSumarWS > ⏸ main >

Output ×

Retriever Output ×   Java DB Database Process ×   GlassFish Server 5.0 ×   Run (TestServicioSumarWS) ×

```
cd C:\Users\Johan\Desktop\consiti 2023\JAVA EE\leccion 10\ClienteSumaWS; "JAVA_HOME=C:\\Program Files\\Java\\jdk1.8.0_202" cmd
Scanning for projects...

--------------------< sv.com.gm.sga:ClienteSumaWS >--------------------
Building ClienteSumaWS 1.0
    from pom.xml
--------------------------------[ jar ]--------------------------------

--- jaxws:2.5:wsimport (default) @ ClienteSumaWS ---

--- resources:3.3.1:resources (default-resources) @ ClienteSumaWS ---
skip non existing resourceDirectory C:\Users\Johan\Desktop\consiti 2023\JAVA EE\leccion 10\ClienteSumaWS\src\main\resources

--- compiler:3.11.0:compile (default-compile) @ ClienteSumaWS ---
Changes detected - recompiling the module! :source
Compiling 8 source files with javac [debug target 1.8] to target\classes

--- exec:3.1.0:exec (default-cli) @ ClienteSumaWS ---
Ejecutando servicio sumar ws
9
Fin Servicio sumar
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  1.540 s
Finished at: 2023-11-03T17:37:18-06:00
------------------------------------------------------------------------
```

SGA web service con jax

```java
package sv.com.gm.sga.servicio;
import java.util.List;
import javax.jws.WebMethod;
import javax.jws.WebService;
import sv.com.gm.sga.domain.Persona;

@WebService
public interface PersonaServiceWS {

    @WebMethod
    public List<Persona> listarPersonas();

}
```

```java
package sv.com.gm.sga.servicio;

import java.util.ArrayList;
import java.util.List;
import javax.annotation.Resource;
import javax.ejb.SessionContext;
import javax.ejb.Stateless;
import javax.inject.Inject;
import javax.jws.WebService;
import sv.com.gm.sga.domain.Persona;
import sv.com.gm.sga.datos.PersonaDao;

@Stateless
@WebService(endpointInterface = "sv.com.gm.sga.servicio.PersonaServiceWS")
public class PersonaServiceImpl implements PersonaServiceRemote, PersonaService, PersonaServiceWS {

    @Inject
```

```java
import java.io.Serializable;
import java.util.List;
import javax.persistence.*;
import javax.validation.constraints.Size;
import javax.xml.bind.annotation.*;

@Entity
@NamedQueries({
    @NamedQuery(name = "Persona.findAll", query = "SELECT p FROM Persona p"),
    @NamedQuery(name = "Persona.findByIdPersona", query = "SELECT p FROM Persona p WHERE p.idPersona = :idPersona"),
    @NamedQuery(name = "Persona.findByNombre", query = "SELECT p FROM Persona p WHERE p.nombre = :nombre"),
    @NamedQuery(name = "Persona.findByApellido", query = "SELECT p FROM Persona p WHERE p.apellido = :apellido"),
    @NamedQuery(name = "Persona.findByEmail", query = "SELECT p FROM Persona p WHERE p.email = :email"),
    @NamedQuery(name = "Persona.findByTelefono", query = "SELECT p FROM Persona p WHERE p.telefono = :telefono")})
@XmlAccessorType(XmlAccessType.FIELD)
public class Persona implements Serializable {

    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

Cliente:

```java
package testclientews;

import clientews.servicio.Persona;
import clientews.servicio.PersonaServiceImplService;
import clientews.servicio.PersonaServiceWS;
import java.util.List;

public class TestPersonaServiceWS {
    public static void main(String[] args) {
        PersonaServiceWS personaservice = new PersonaServiceImplService().getPersonaServiceImplPort();
        System.out.println(x: "ejecutando servicio de listar personas ws");
        List<Persona> personas = personaservice.listarPersonas();
        for(Persona persona: personas){
            System.out.println("Persona idPersona:" + persona.getIdPersona() + ", nombre:" + persona.getNombre()
            + ", apellido:" +  persona.getApellido() + ", email:" + persona.getEmail());
        }
        System.out.println(x: "Fin servicio listar personas ws");

    }
}
```

Find: [   ] [v] Previous Next Select aA „" .*

testclientews.TestPersonaServiceWS > main >

Output ×

Retriever Output ×   Java DB Database Process ×   GlassFish Server 5.0 ×   Run (TestPersonaServiceWS) ×

```
Compiling 9 source files with javac [debug target 1.8] to target\classes

--- exec:3.1.0:exec (default-cli) @ ClienteSgaWS ---
ejecutando servicio de listar personas ws
Persona idPersona:1, nombre:Juan, apellido:Higuain, email:jperez@mail.com
Persona idPersona:2, nombre:Karla, apellido:Gomez, email:kgomez@mail.com
Persona idPersona:3, nombre:Maria, apellido:Gutierrez, email:mgutierrez@mail.com
Persona idPersona:4, nombre:Pedro, apellido:Luna, email:pedro@gmail.com
Persona idPersona:6, nombre:Hugo, apellido:Hernandez, email:hhernandez@mail.com
Persona idPersona:7, nombre:pepe, apellido:problemas, email:pepe@hotmail.com
Fin servicio listar personas ws
```