

# PLAN

23: SOAP Web Services - 2

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni

### WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

### Invocare un WS

Un esempio riassuntivo

### Supporto ai WS in Netbeans

Il progetto per WS

Testing

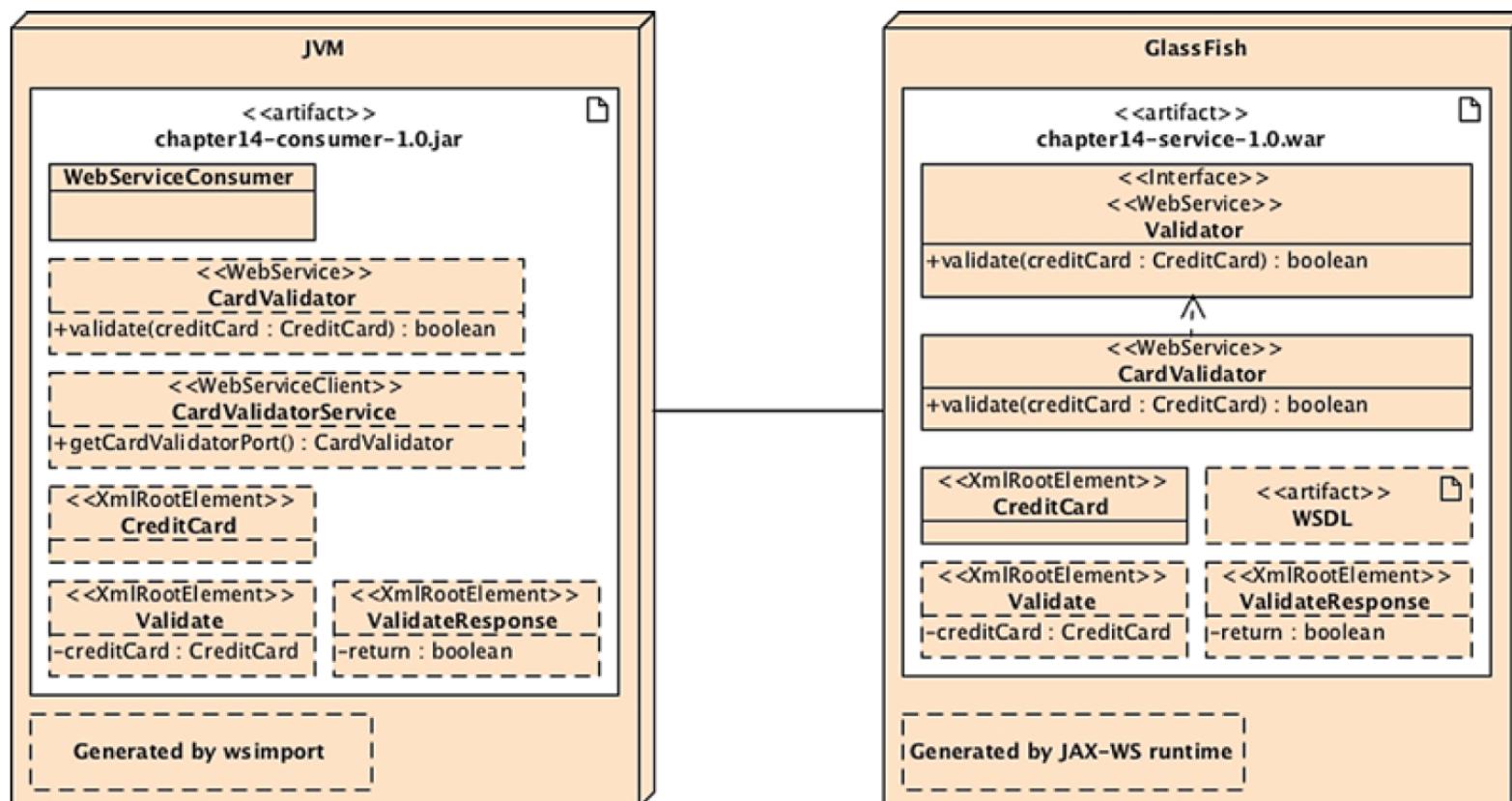
WS Client

### Conclusioni



# DIAGRAMMA DELL'ESEMPIO

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# LA CLASSE PER LA CARTA DI CREDITO

23: SOAP Web Services - 2

```
@XmlRootElement<--  
@XmlAccessorType(XmlAccessType.FIELD)  
public class CreditCard {  
  
    @XmlAttribute(required = true)  
    private String number;  
  
    @XmlAttribute(name = "expiry_date", required = true)  
    private String expiryDate;  
  
    @XmlAttribute(name = "control_number", required = true)  
    private Integer controlNumber;  
  
    @XmlAttribute(required = true)  
    private String type;  
  
    // Constructors, getters, setters  
}
```

Inizio dell'elemento root dell'XML

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# LA CLASSE PER LA CARTA DI CREDITO

23: SOAP Web Services - 2

```
@XmlRootElement  
@XmlAccessorType(XmlAccessType.FIELD) ←  
public class CreditCard {  
  
    @XmlAttribute(required = true)  
    private String number;  
  
    @XmlAttribute(name = "expiry_date", required = true)  
    private String expiryDate;  
  
    @XmlAttribute(name = "control_number", required = true)  
    private Integer controlNumber;  
  
    @XmlAttribute(required = true)  
    private String type;  
  
    // Constructors, getters, setters  
}
```

Inizio dell'elemento root dell'XML

Tutti i campi saranno mappati su XML

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# LA CLASSE PER LA CARTA DI CREDITO

23: SOAP Web Services - 2

```
@XmlRootElement  
@XmlAccessorType(XmlAccessType.FIELD)  
public class CreditCard {  
  
    @XmlAttribute(required = true)  
    private String number;  
  
    @XmlAttribute(name = "expiry_date", required = true)  
    private String expiryDate;  
  
    @XmlAttribute(name = "control_number", required = true)  
    private Integer controlNumber;  
  
    @XmlAttribute(required = true)  
    private String type;  
  
    // Constructors, getters, setters  
}
```

- ▶ Inizio dell'elemento root dell'XML
- ▶ Tutti i campi saranno mappati su XML
- ▶ **Attributo obbligatorio**

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# LA CLASSE PER LA CARTA DI CREDITO

23: SOAP Web Services - 2

```
@XmlRootElement  
@XmlAccessorType(XmlAccessType.FIELD)  
public class CreditCard {  
  
    @XmlAttribute(required = true)  
    private String number;  
  
    @XmlAttribute(name = "expiry_date", required = true) ←  
    private String expiryDate;  
  
    @XmlAttribute(name = "control_number", required = true)  
    private Integer controlNumber;  
  
    @XmlAttribute(required = true)  
    private String type;  
  
    // Constructors, getters, setters  
}
```

- ▶ Inizio dell'elemento root dell'XML
- ▶ Tutti i campi saranno mappati su XML
- ▶ Attributo obbligatorio
- ▶ Attributo obbligatorio, con il nome XML diverso

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# LA CLASSE PER LA CARTA DI CREDITO

23: SOAP Web Services - 2

```
@XmlRootElement  
@XmlAccessorType(XmlAccessType.FIELD)  
public class CreditCard {  
  
    @XmlAttribute(required = true)  
    private String number;  
  
    @XmlAttribute(name = "expiry_date", required = true)  
    private String expiryDate;  
  
    @XmlAttribute(name = "control_number", required = true) ←  
    private Integer controlNumber;  
  
    @XmlAttribute(required = true)  
    private String type;  
  
    // Constructors, getters, setters  
}
```

- ▶ Inizio dell'elemento root dell'XML
- ▶ Tutti i campi saranno mappati su XML
- ▶ Attributo obbligatorio
- ▶ Attributo obbligatorio, con il nome XML diverso
- ▶ Attributo obbligatorio, con il nome XML diverso

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# LA CLASSE PER LA CARTA DI CREDITO

23: SOAP Web Services - 2

```
@XmlRootElement  
@XmlAccessorType(XmlAccessType.FIELD)  
public class CreditCard {  
  
    @XmlAttribute(required = true)  
    private String number;  
  
    @XmlAttribute(name = "expiry_date", required = true)  
    private String expiryDate;  
  
    @XmlAttribute(name = "control_number", required = true)  
    private Integer controlNumber;  
  
    @XmlAttribute(required = true) ←  
    private String type;  
  
    // Constructors, getters, setters  
}
```

- ▶ Inizio dell'elemento root dell'XML
- ▶ Tutti i campi saranno mappati su XML
- ▶ Attributo obbligatorio
- ▶ Attributo obbligatorio, con il nome XML diverso
- ▶ Attributo obbligatorio, con il nome XML diverso
- ▶ Attributo obbligatorio

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# IL WEBSERVICE

23: SOAP Web Services - 2

```
@WebService
public interface Validator {
    public boolean validate(CreditCard creditCard);
}

@WebService(endpointInterface =
        "org.agoncal.book.javaee7.chapter14.Validator")
public class CardValidator implements Validator {
    public boolean validate(CreditCard creditCard) {
        Character lastDigit = creditCard.getNumber().charAt(
                creditCard.getNumber().length() - 1);
        if (Integer.parseInt(lastDigit.toString()) % 2 == 0) {
            return true;
        } else {
            return false;
        }
    }
}
```

Con annotazione da WS...

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# IL WEBSERVICE

23: SOAP Web Services - 2

```
@WebService
public interface Validator {
    public boolean validate(CreditCard creditCard);
}

@WebService(endpointInterface =
        "org.agoncal.book.javaee7.chapter14.Validator")
public class CardValidator implements Validator {
    public boolean validate(CreditCard creditCard) {
        Character lastDigit = creditCard.getNumber().charAt(
                creditCard.getNumber().length() - 1);
        if (Integer.parseInt(lastDigit.toString()) % 2 == 0) {
            return true;
        } else {
            return false;
        }
    }
}
```

Con annotazione da WS...  
... si dichiara l'interfaccia

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# IL WEBSERVICE

23: SOAP Web Services - 2

```
@WebService
public interface Validator {
    public boolean validate(CreditCard creditCard);
}

@WebService(endpointInterface =
        "org.agoncal.book.javaee7.chapter14.Validator")
public class CardValidator implements Validator {
    public boolean validate(CreditCard creditCard) {
        Character lastDigit = creditCard.getNumber().charAt(
                creditCard.getNumber().length() - 1);
        if (Integer.parseInt(lastDigit.toString()) % 2 == 0) {
            return true;
        } else {
            return false;
        }
    }
}
```

- ▶ Con annotazione da WS...
- ▶ ... si dichiara l'interfaccia
- ▶ Poi si dichiara il WS

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# IL WEBSERVICE

23: SOAP Web Services - 2

```
@WebService
public interface Validator {
    public boolean validate(CreditCard creditCard);
}

@WebService(endpointInterface =
        "org.agoncal.book.javaee7.chapter14.Validator")
public class CardValidator implements Validator {
    public boolean validate(CreditCard creditCard) {
        Character lastDigit = creditCard.getNumber().charAt(
                creditCard.getNumber().length() - 1);
        if (Integer.parseInt(lastDigit.toString()) % 2 == 0) {
            return true;
        } else {
            return false;
        }
    }
}
```

- ▶ Con annotazione da WS...
  - ▶ ... si dichiara l'interfaccia
  - ▶ Poi si dichiara il WS
- Con la classe che implementa l'interfaccia

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# IL WEBSERVICE

23: SOAP Web Services - 2

```
@WebService  
public interface Validator {  
    public boolean validate(CreditCard creditCard);  
}  
  
@WebService(endpointInterface =  
            "org.agoncal.book.javaee7.chapter14.Validator")  
public class CardValidator implements Validator {  
    public boolean validate(CreditCard creditCard) {  
        Character lastDigit = creditCard.getNumber().charAt(  
            creditCard.getNumber().length() - 1);  
        if (Integer.parseInt(lastDigit.toString()) % 2 == 0) {  
            return true;  
        } else {  
            return false;  
        }  
    }  
}
```

- ▶ Con annotazione da WS...
- ▶ ... si dichiara l'interfaccia
- ▶ Poi si dichiara il WS
- ▶ Con la classe che implementa l'interfaccia
- ▶ **Metodo offerto come WS**

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# PLAN

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni

23: SOAP Web Services - 2

### WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

### Invocare un WS

Un esempio riassuntivo

### Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

### Conclusioni



# PLAN

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni

23: SOAP Web Services - 2

### WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

### Invocare un WS

Un esempio riassuntivo

### Supporto ai WS in Netbeans

Il progetto per WS

Testing

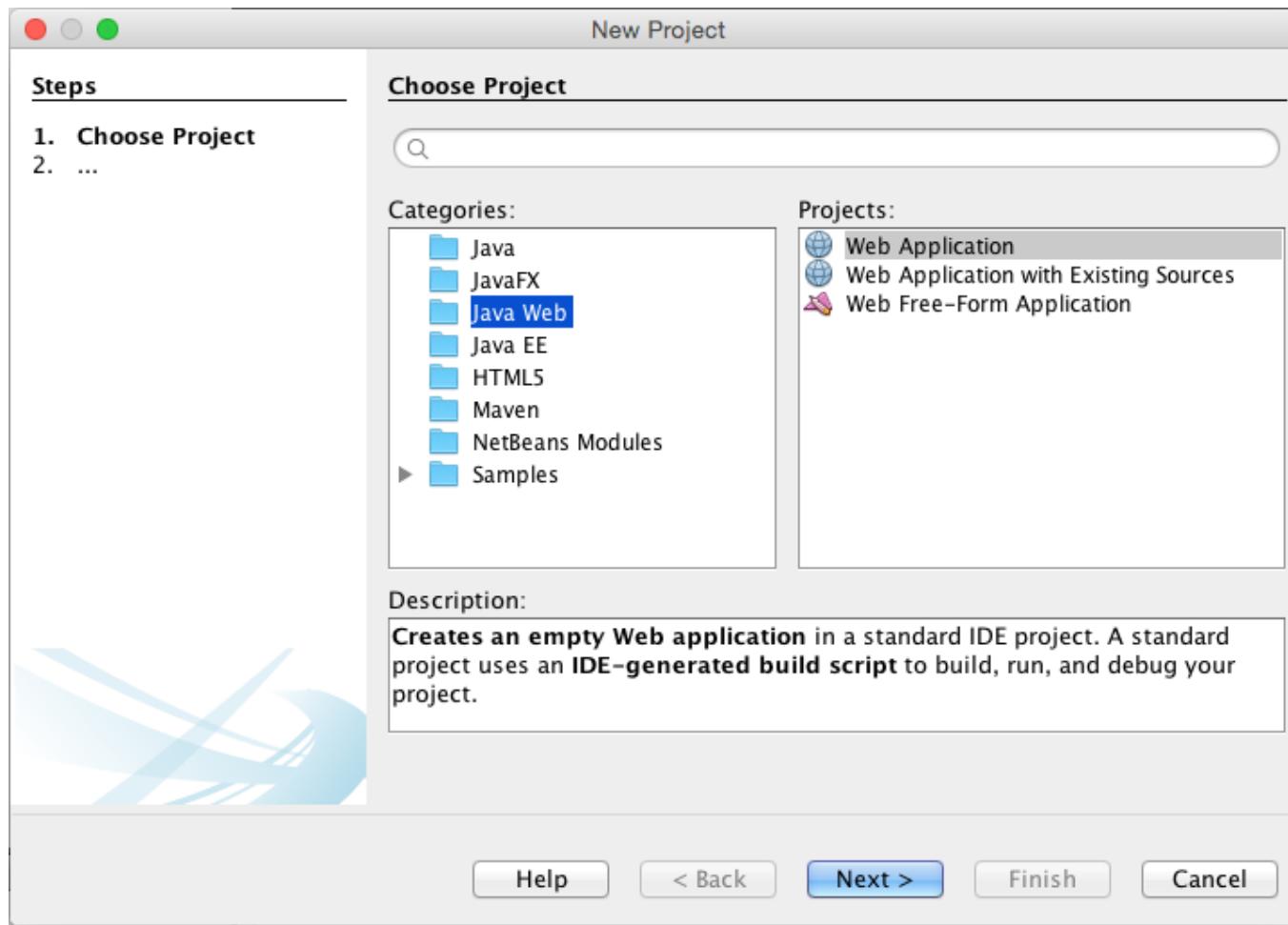
WS Client

### Conclusioni



# WS IN NETBEANS: CREAZIONE DI UN PROGETTO PER WS

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

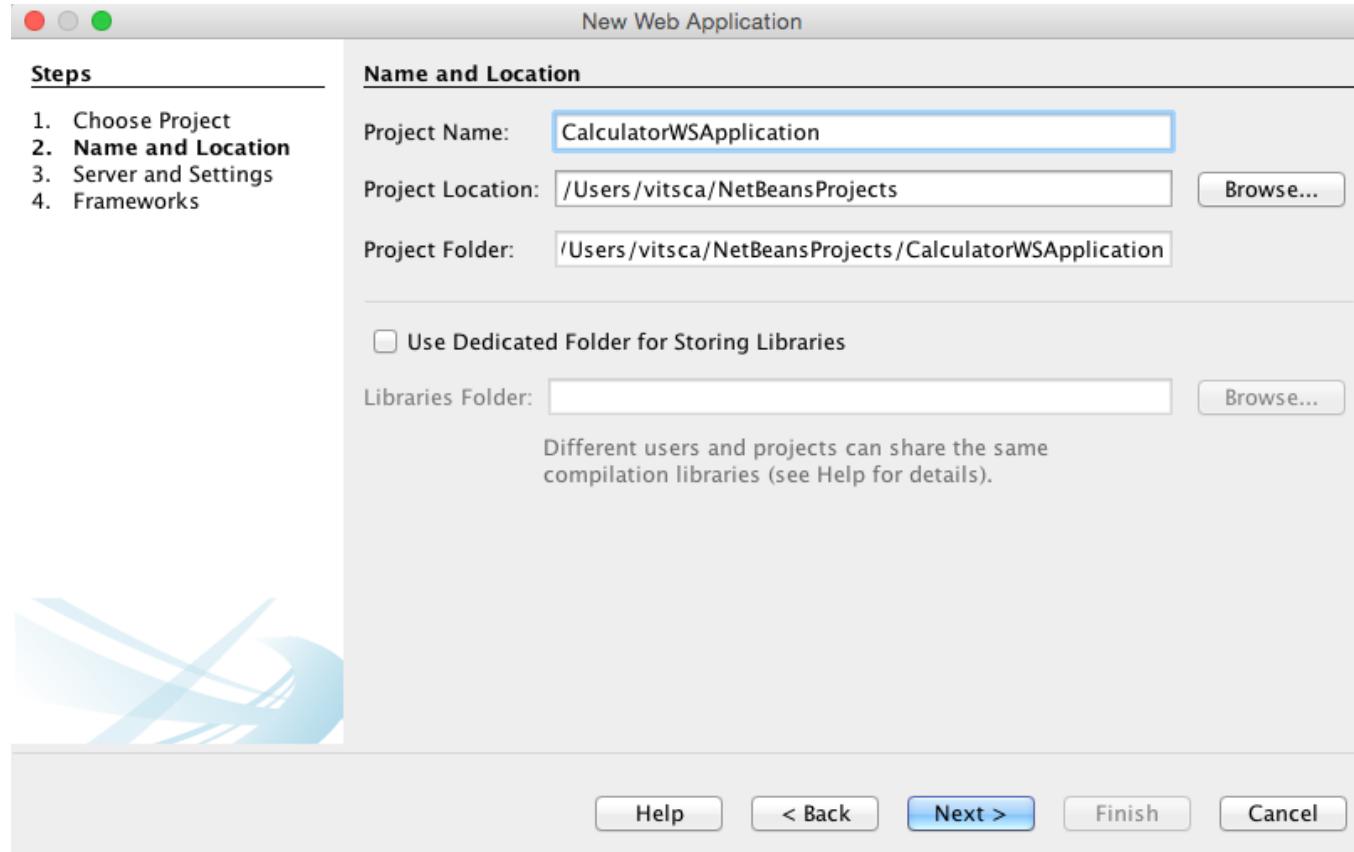
WS Client

Conclusioni



# WS IN NETBEANS: CREAZIONE PROGETTO - 1

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

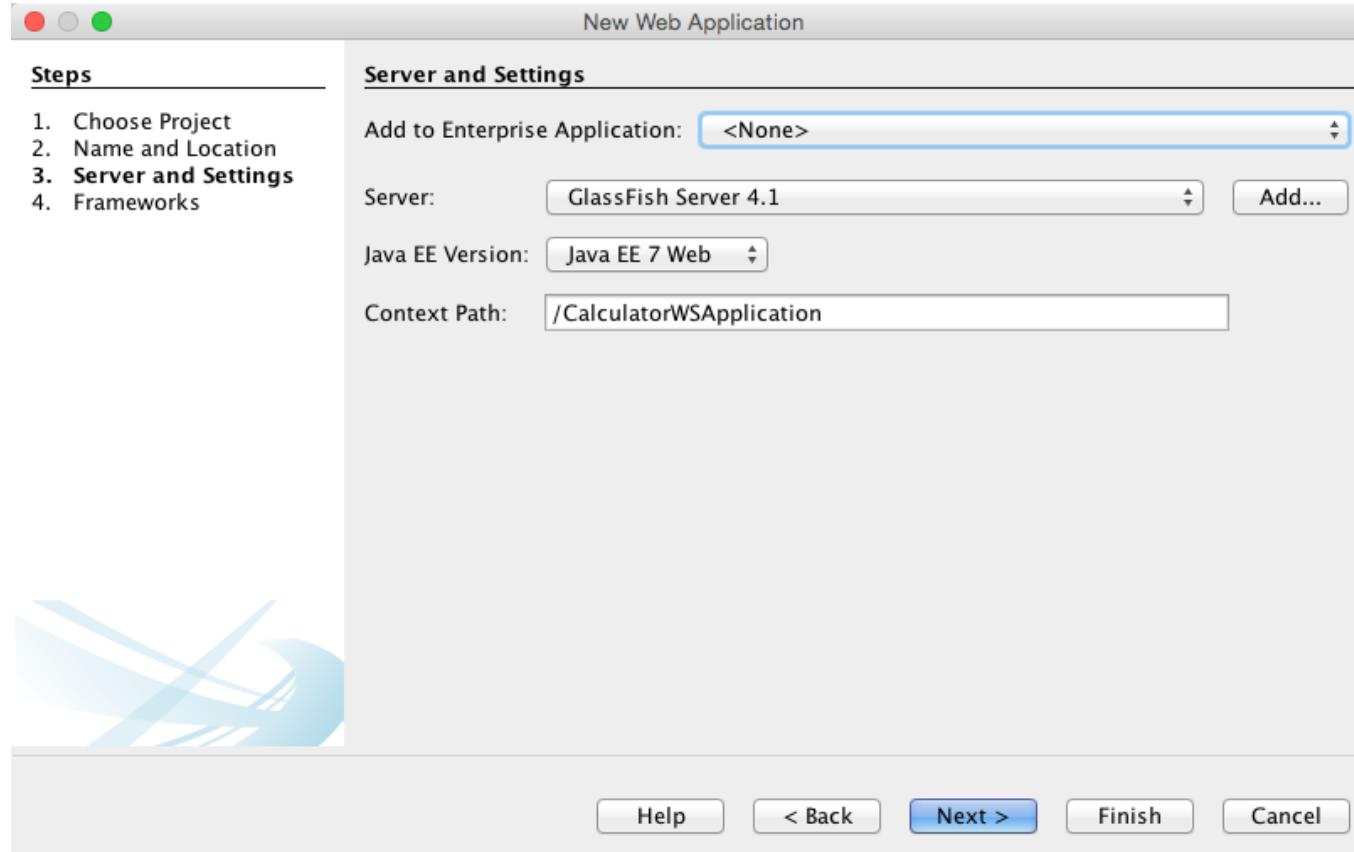
WS Client

Conclusioni



# WS IN NETBEANS: CREAZIONE PROGETTO - 2

23: SOAP Web Services - 2



## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

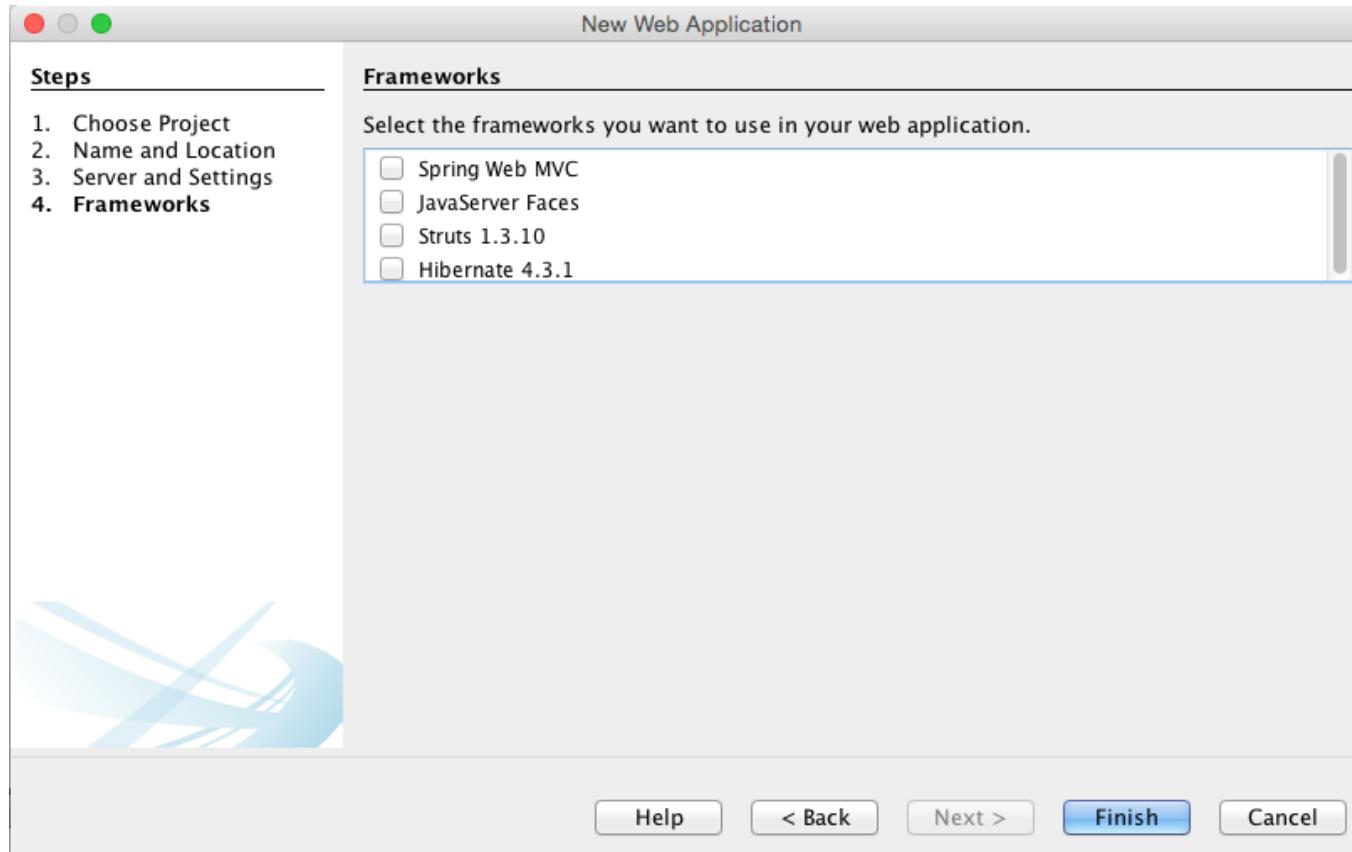
WS Client

## Conclusioni



# WS IN NETBEANS: CREAZIONE PROGETTO - 3

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

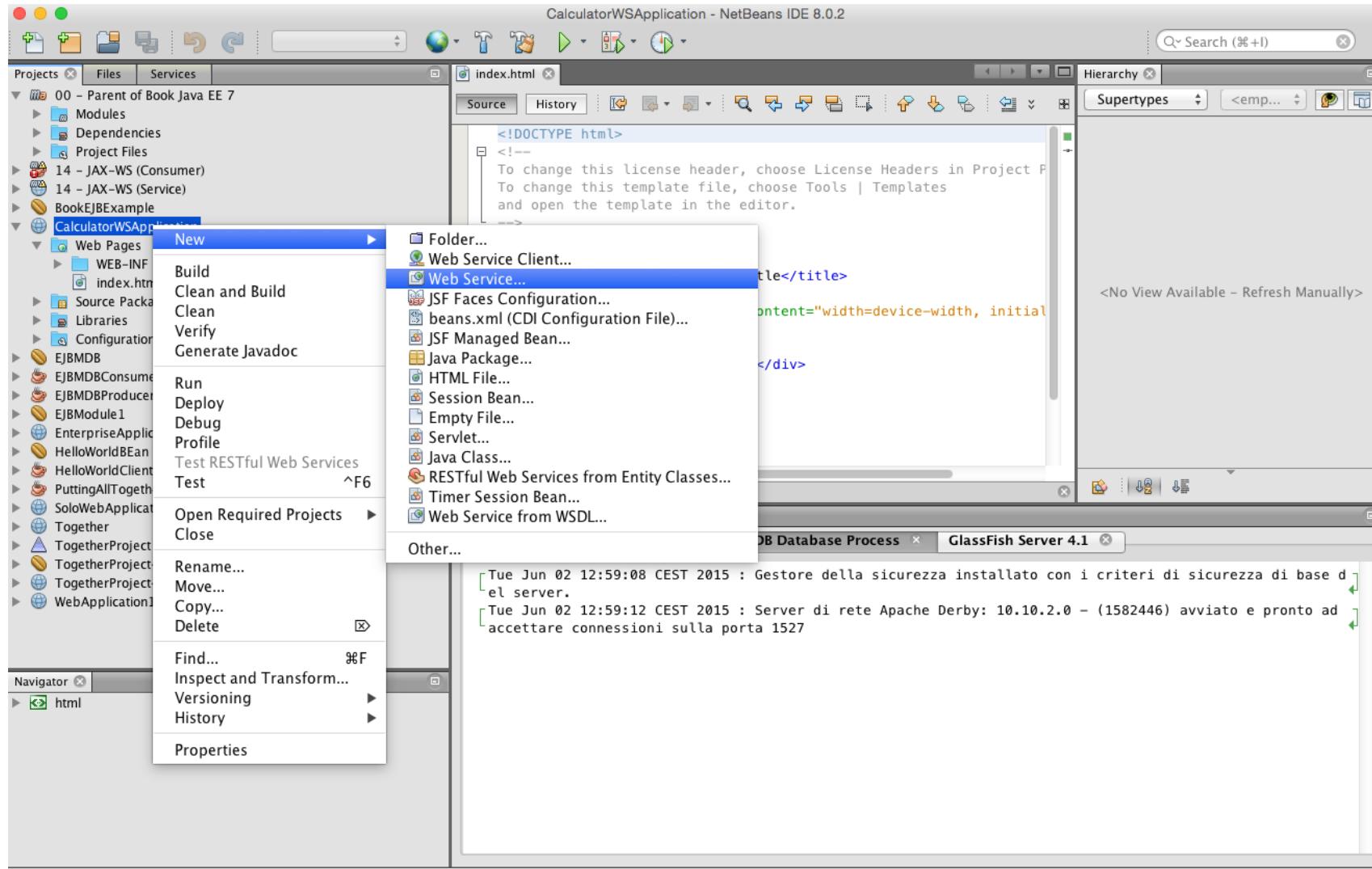
Testing

WS Client

Conclusioni



# WS IN NETBEANS: CREAZIONE DI UN WS - 1



## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# WS IN NETBEANS: CREAZIONE DI UN WS - 2

23: SOAP Web Services - 2

New Web Service

**Steps**

1. Choose File Type
2. Name and Location

**Name and Location**

Web Service Name:

Project:

Location:

Package:

Create Web Service from Scratch

Create Web Service from Existing Session Bean

Enterprise Bean:

Implement Web Service as Stateless Session Bean

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

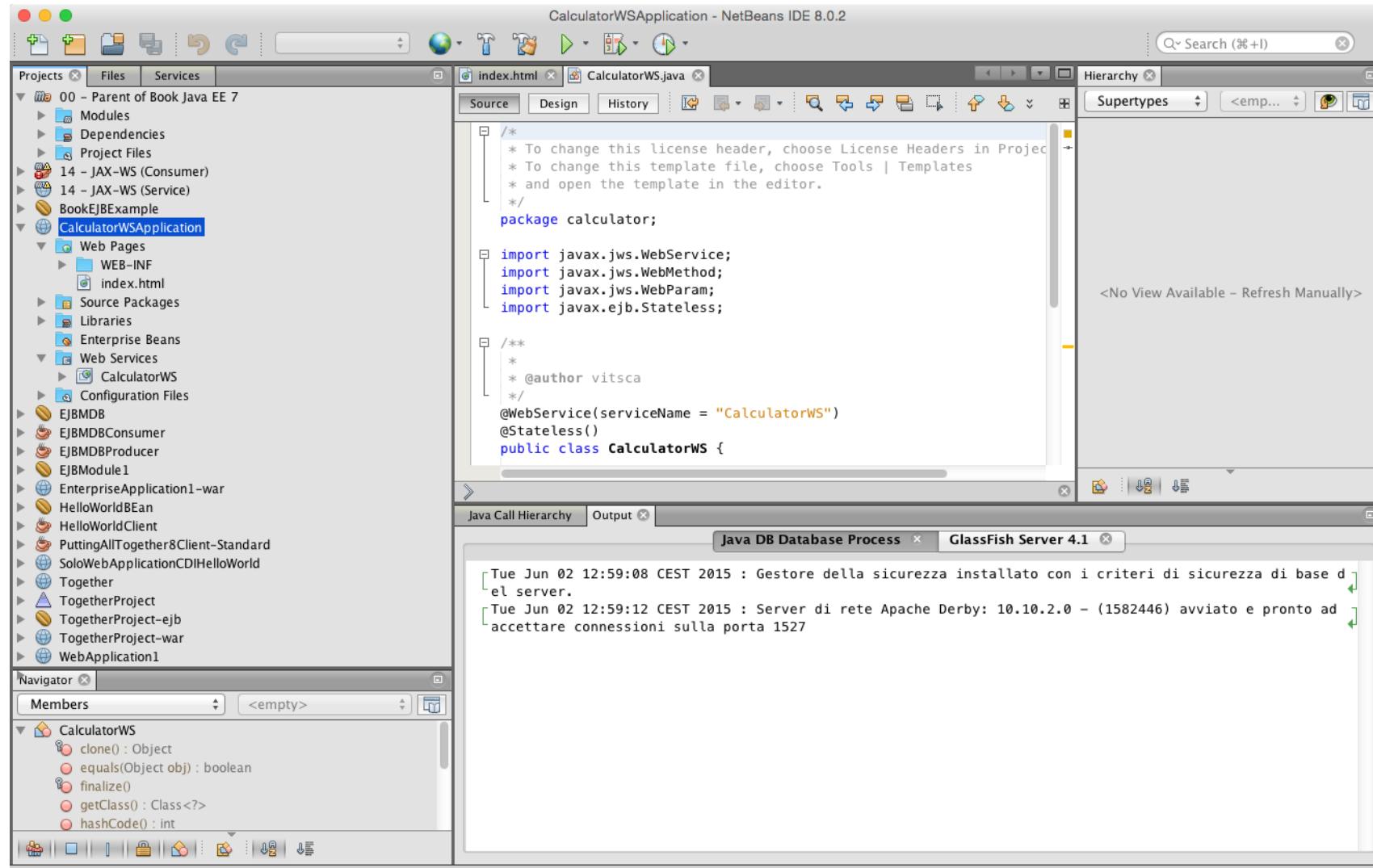
Testing

WS Client

Conclusioni



# WS IN NETBEANS: LA SITUAZIONE



## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# WS IN NETBEANS: UN WS DI HELLO

The screenshot shows the NetBeans IDE interface with the following details:

- Project Explorer:** Shows a single file named "CalculatorWS.java".
- Toolbars:** Standard NetBeans toolbars for file operations, search, and navigation.
- Code Editor:** Displays the Java code for the "CalculatorWS" class. The code includes annotations for a WebService and a WebMethod, and a simple implementation of the "hello" method.

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package calculator;

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

/**
 *
 * @author vitsca
 */
@WebService(serviceName = "CalculatorWS")
@Stateless()
public class CalculatorWS {

    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "hello")
    public String hello(@WebParam(name = "name") String txt) {
        return "Hello " + txt + " !";
    }
}
```

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# WS IN NETBEANS: VISTA DI DESIGN

The screenshot shows the NetBeans IDE interface for designing a Web Service. The title bar indicates the project is 'CalculatorWS.java'. The main window displays the 'Design' tab selected. The 'Operations' section shows one operation named 'hello' with a single parameter named 'name' of type 'java.lang.String'. The 'Quality Of Service' section contains three checkboxes: 'Optimize Transfer Of Binary Data (MTOM)', 'Reliable Message Delivery', and 'Secure Service'. An 'Edit Web Service Attributes...' button is located at the bottom right of this panel.

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

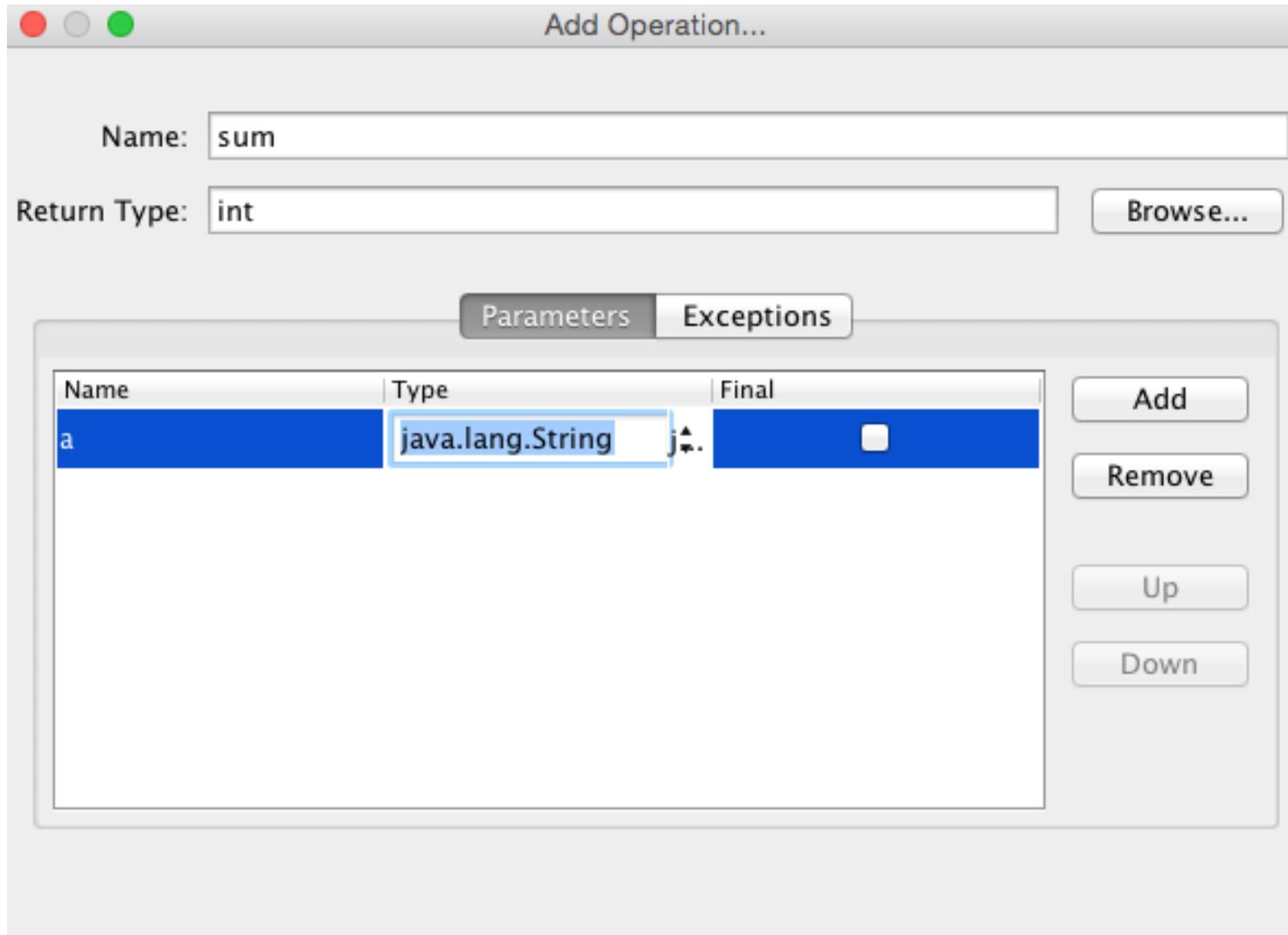
Testing

WS Client

Conclusioni



# WS IN NETBEANS: AGGIUNGIAMO UNA OPERAZIONE



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# WS IN NETBEANS: UN PARAMETRO ...

Add Operation...

Name: sum

Return Type: int

| Name | Type             | Final                    |
|------|------------------|--------------------------|
| a    | java.lang.String | <input type="checkbox"/> |
|      | boolean          |                          |
|      | int              |                          |
|      | char             |                          |
|      | byte             |                          |
|      | short            |                          |
|      | long             |                          |
|      | float            |                          |
|      | double           |                          |

**WS in Java**

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

**Invocare un WS**

Un esempio riassuntivo

**Supporto ai WS in Netbeans**

Il progetto per WS

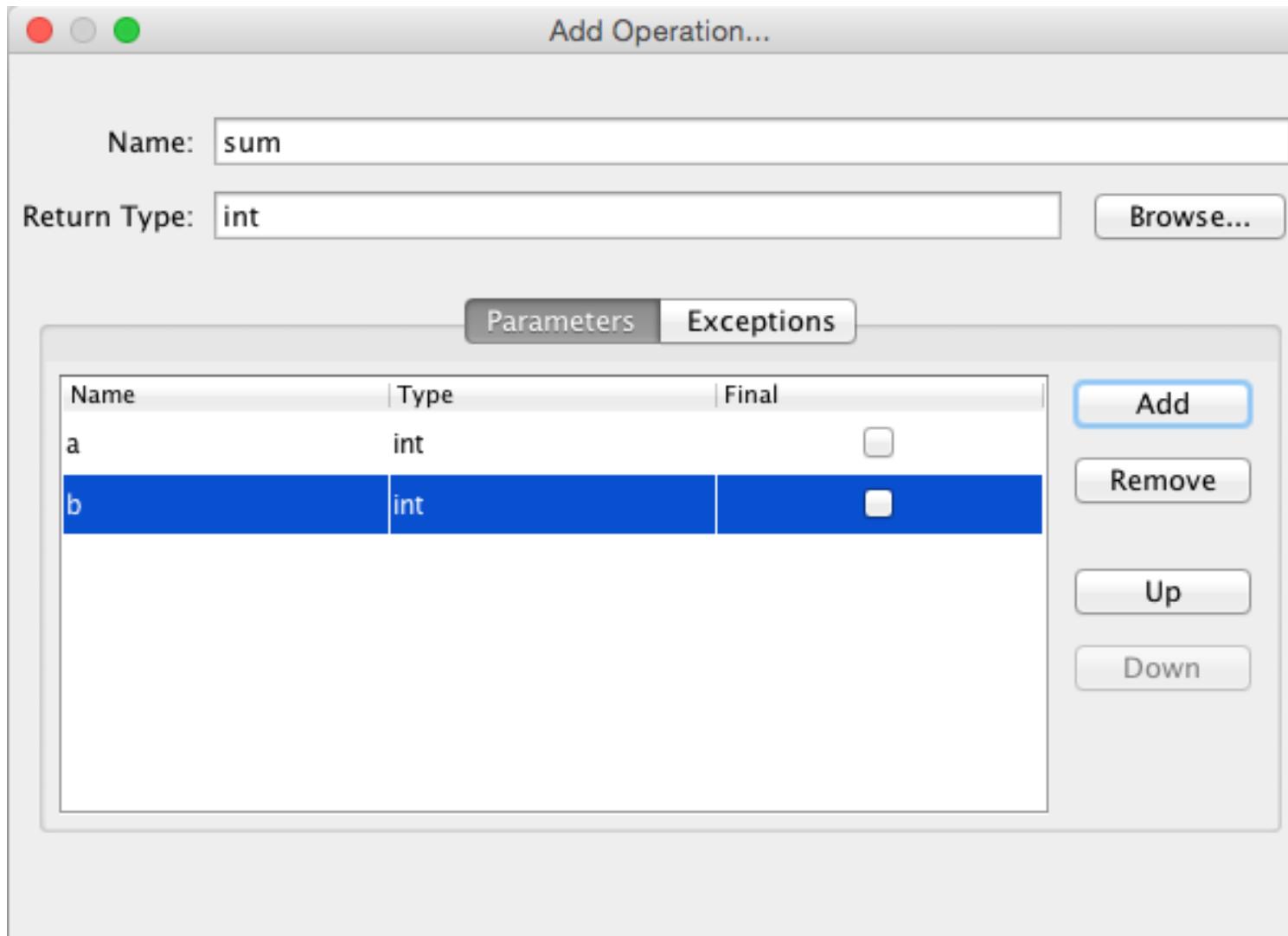
Testing

WS Client

**Conclusioni**



# WS IN NETBEANS: UN SECONDO PARAMETRO



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# WS IN NETBEANS: LA VISTA DI DESIGN

index.html    CalculatorWS.java

Source    Design    History    100%    Add    Remove    Settings

CalculatorWS

**Operations (2)**

Add Operation...    Remove Operation

| hello      |                        | Output | Faults | Description                        |
|------------|------------------------|--------|--------|------------------------------------|
| Parameters | Parameter Name<br>name |        |        | Parameter Type<br>java.lang.String |

| sum        |                          | Output | Faults | Description                  |
|------------|--------------------------|--------|--------|------------------------------|
| Parameters | Parameter Name<br>a<br>b |        |        | Parameter Type<br>int<br>int |

**Quality Of Service**

- Optimize Transfer Of Binary Data (MTOM)
- Reliable Message Delivery
- Secure Service

Edit Web Service Attributes...

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# WS IN NETBEANS: IL CODICE

The screenshot shows the NetBeans IDE interface with the following details:

- Project Explorer:** Shows the file `CalculatorWS.java` selected.
- Code Editor:** Displays the Java code for the `CalculatorWS` class. The code includes annotations for web methods and parameters, such as `@WebService(serviceName = "CalculatorWS")`, `@WebMethod(operationName = "hello")`, and `@WebParam(name = "name")`.
- Toolbars:** Standard NetBeans toolbars for navigation and file operations.
- Status Bar:** Shows the status "42/74" and a small circular icon with a diagonal line.

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

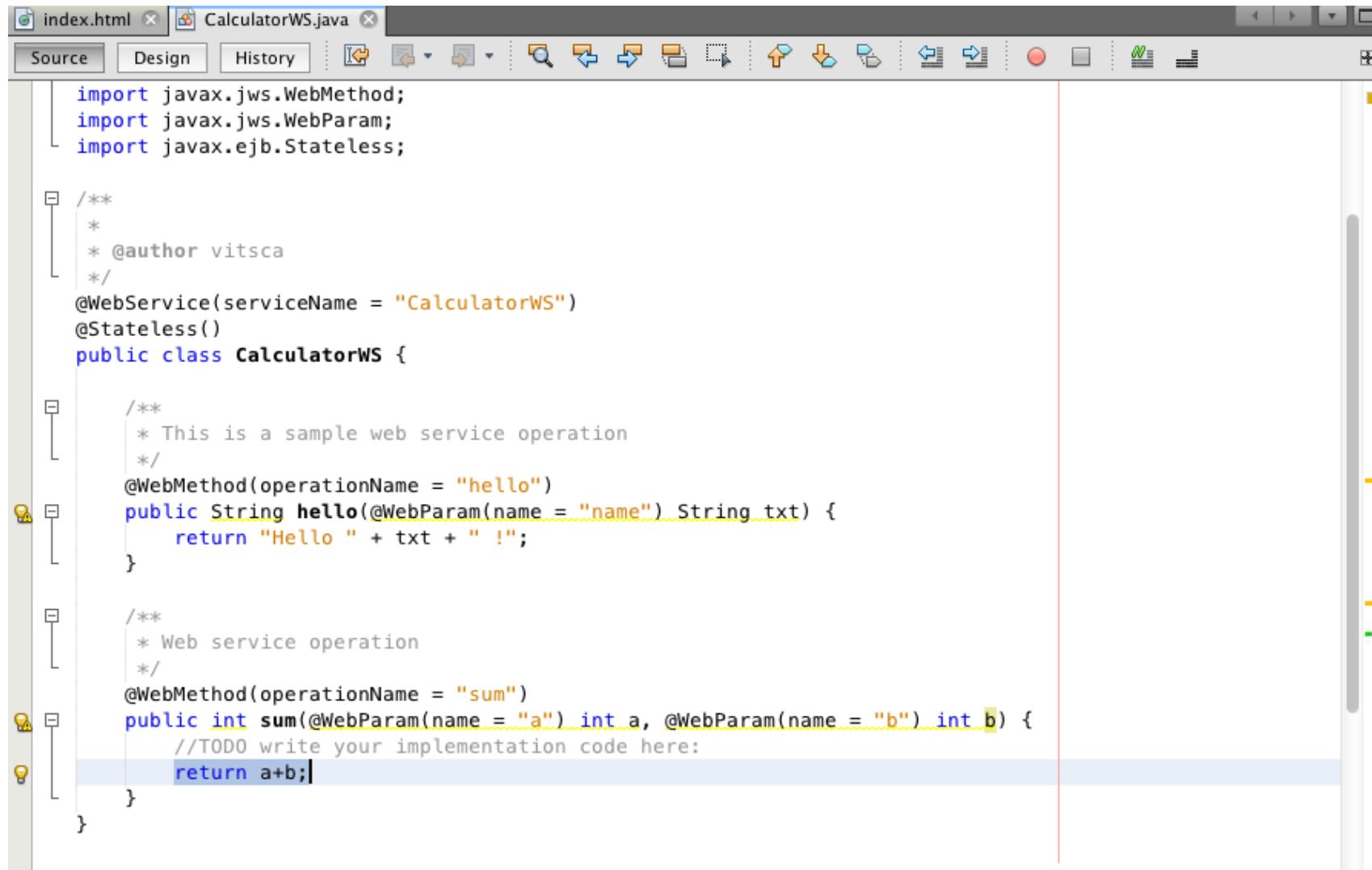
Testing

WS Client

## Conclusioni



# WS IN NETBEANS: SCRIVIAMO QUALCOSA!



```

import javax.jws.WebMethod;
import javax.jws.WebParam;
import javax.ejb.Stateless;

/**
 * 
 * @author vitsca
 */
@WebService(serviceName = "CalculatorWS")
@Stateless()
public class CalculatorWS {

    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "hello")
    public String hello(@WebParam(name = "name") String txt) {
        return "Hello " + txt + " !";
    }

    /**
     * Web service operation
     */
    @WebMethod(operationName = "sum")
    public int sum(@WebParam(name = "a") int a, @WebParam(name = "b") int b) {
        //TODO write your implementation code here:
        return a+b;
    }
}

```

## WS in Java

[WSDL Mapping](#)

[Eccezioni e Fault](#)

[Contesto e ciclo di vita](#)

## Invocare un WS

[Un esempio riassuntivo](#)

## Supporto ai WS in Netbeans

[Il progetto per WS](#)

[Testing](#)

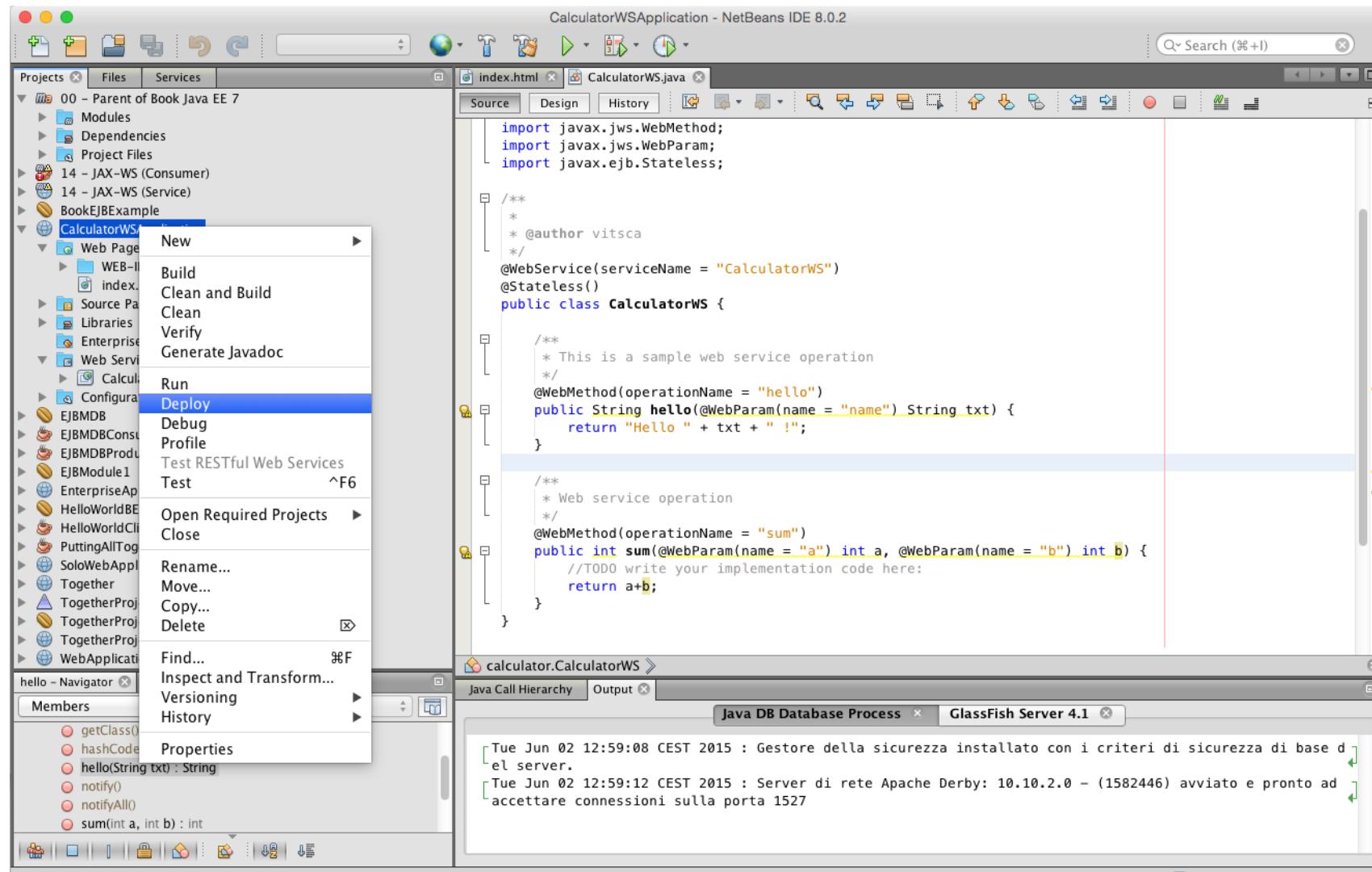
[WS Client](#)

## Conclusioni



# WS IN NETBEANS: DEPLOY

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni

# WS IN NETBEANS: SUL SERVER

Java Call Hierarchy   Output

Java DB Database Process GlassFish Server 4.1 CalculatorWSApplication (run-deploy)

```

Avvertenza: Flashlight listener registration failed for listener class: com.sun.ejb.monitoring.stats.StatelessSessionBeanStatsProvider , will retry later
WARN: WELD-000411: Observer method [BackedAnnotatedMethod] public org.glassfish.jms.injection.JMSCDIExtension.processAnnotatedType(@Observes ProcessAnnotatedType<Object>) receives events for all annotated types. Consider restricting events using @WithAnnotations or a generic type with bounds.
WARN: WELD-000411: Observer method [BackedAnnotatedMethod] org.glassfish.sse.impl.ServerSentEventCDIExtension.processAnnotatedType(@Observes ProcessAnnotatedType<Object>, BeanManager) receives events for all annotated types. Consider restricting events using @WithAnnotations or a generic type with bounds.
WARN: WELD-000411: Observer method [BackedAnnotatedMethod] private org.glassfish.jersey.gf.cdi.internal.CdiComponentProvider.processAnnotatedType(@Observes ProcessAnnotatedType<Object>) receives events for all annotated types. Consider restricting events using @WithAnnotations or a generic type with bounds.
Informazioni: EJB Endpoint deployed CalculatorWSApplication
  listening at address at http://MacBook-Pro-di-Vittorio.local:8080/CalculatorWS/CalculatorWS
Informazioni: EJB Endpoint deployed CalculatorWSApplication
  listening at address at http://MacBook-Pro-di-Vittorio.local:8080/CalculatorWS/CalculatorWS
Informazioni: Loading application [CalculatorWSApplication] at [/CalculatorWSApplication]
Informazioni: CalculatorWSApplication was successfully deployed in 2.436 milliseconds.

```

## WS in Java

[WSDL Mapping](#)

[Eccezioni e Fault](#)

[Contesto e ciclo di vita](#)

## Invocare un WS

[Un esempio riassuntivo](#)

## Supporto ai WS in Netbeans

[Il progetto per WS](#)

[Testing](#)

[WS Client](#)

## Conclusioni



# WS IN NETBEANS: AL DEPLOY COSA SI FA

```
Java Call Hierarchy Output Java DB Database Process | GlassFish Server 4.1 | CalculatorWSApplication (run-deploy) x
/LUsers/vitsca/NetBeansProjects/CalculatorWSApplication run-deploy
init:
deps-module-jar:
deps-ear-jar:
deps-jar:
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSApplication/build/web/WEB-INF/classes
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSApplication/build/web/META-INF
Copying 1 file to /Users/vitsca/NetBeansProjects/CalculatorWSApplication/build/web/META-INF
Copying 2 files to /Users/vitsca/NetBeansProjects/CalculatorWSApplication/build/web
library-inclusion-in-archive:
library-inclusion-in-manifest:
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSApplication/build/empty
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSApplication/build/generated-sources/ap-sourc
e-output
compile:
compile-jsps:
In-place deployment at /Users/vitsca/NetBeansProjects/CalculatorWSApplication/build/web
run-deploy:
BUILD SUCCESSFUL (total time: 2 seconds)
```

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# PLAN

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni

23: SOAP Web Services - 2

### WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

### Invocare un WS

Un esempio riassuntivo

### Supporto ai WS in Netbeans

Il progetto per WS

Testing

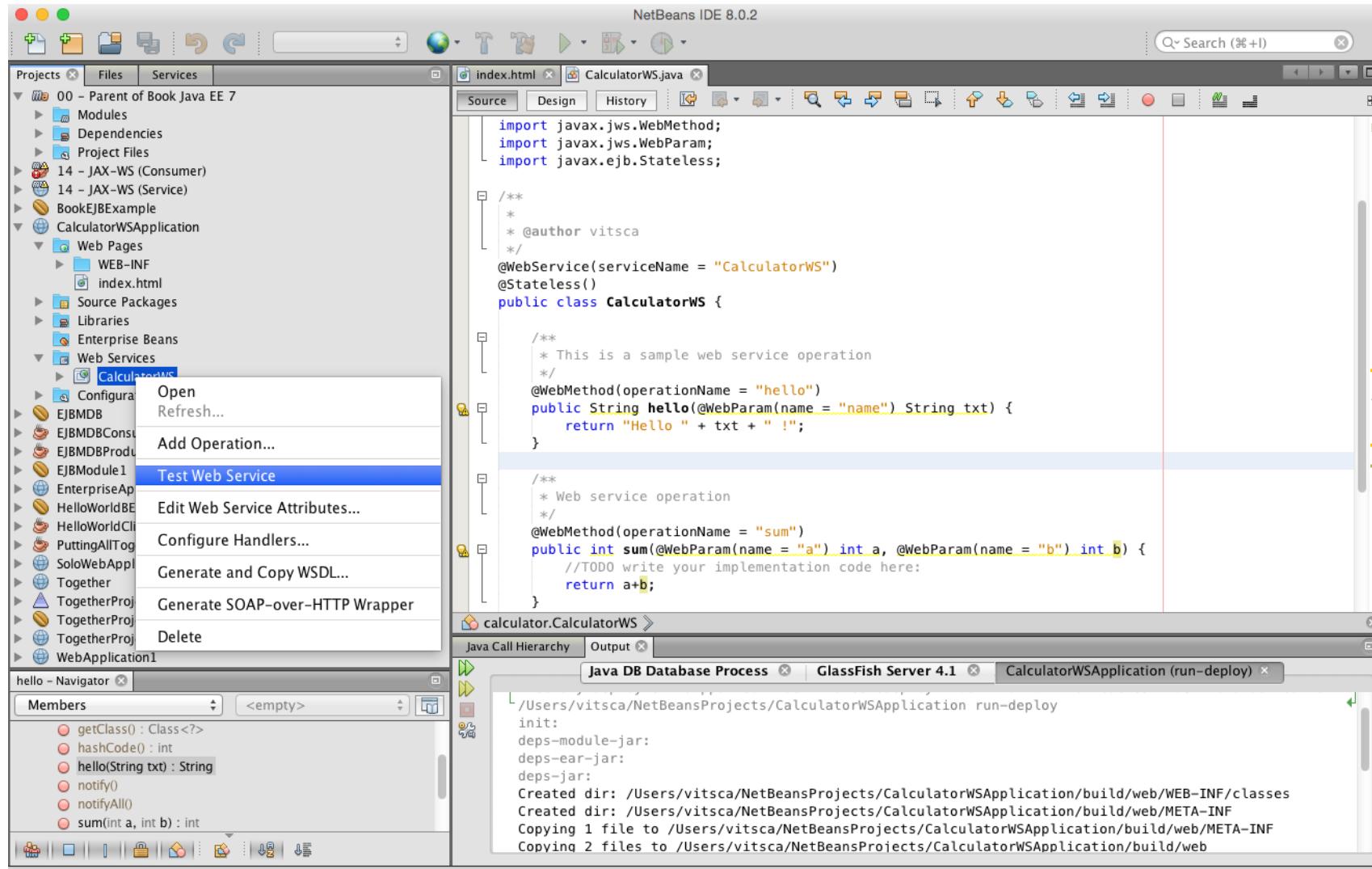
WS Client

### Conclusioni



# WS IN NETBEANS: TEST - 1

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# WS IN NETBEANS: TEST - 2

23: SOAP Web Services - 2

The screenshot shows a web browser window titled "localhost" displaying the "CalculatorWS Web Service Tester". The page header reads "CalculatorWS Web Service Tester". Below it, a message says: "This form will allow you to test your web service implementation ([WSDL File](#))". A note below states: "To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name." A section titled "Methods :" lists two methods:

- sum** (  ,  )
- hello** (  )

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# WS IN NETBEANS: TEST - 3

23: SOAP Web Services - 2

The screenshot shows the 'CalculatorWS Web Service Tester' window from NetBeans. The title bar says 'localhost'. The main content area has a heading 'CalculatorWS Web Service Tester'. Below it, a message says 'This form will allow you to test your web service implementation ([WSDL File](#))'. A note below says 'To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.' A section titled 'Methods :' lists two methods:

**Methods :**

public abstract int calculator.CalculatorWS.sum(int,int)  
sum (  ,  )

public abstract java.lang.String calculator.CalculatorWS.hello(java.lang.String)  
hello (  )

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

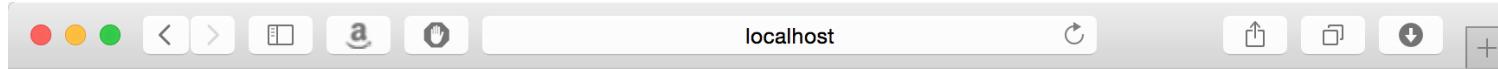
WS Client

Conclusioni



# WS IN NETBEANS: TEST - 4

23: SOAP Web Services - 2



## sum Method invocation

### Method parameter(s)

| Type | Value |
|------|-------|
| int  | 3     |
| int  | 5     |

### Method returned

int : "8"

### SOAP Request

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

### SOAP Response

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# WS IN NETBEANS: TEST - 5

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

<!--
  Published by JAX-WS RI (http://jax-ws.java.net). RI's version is Metro/2.3.1-b419 (branches/2.3.1.x-7937;
-->
<!--
  Generated by JAX-WS RI (http://jax-ws.java.net). RI's version is Metro/2.3.1-b419 (branches/2.3.1.x-7937;
-->
<definitions xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
  xmlns: wsp="http://www.w3.org/ns/ws-policy" xmlns: wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns: wsam="http://www.w3.org/2007/05/addressing/metadata"
  xmlns: soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns: tns="http://calculator/"
  xmlns: xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"
  targetNamespace="http://calculator/" name="CalculatorWS">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://calculator/"
        schemaLocation="http://localhost:8080/CalculatorWS/CalculatorWS?xsd=1"/>
    </xsd:schema>
  </types>
  <message name="sum">
    <part name="parameters" element="tns:sum"/>
  </message>
  <message name="sumResponse">
    <part name="parameters" element="tns:sumResponse"/>
  </message>
  <message name="hello">
    <part name="parameters" element="tns:hello"/>
  </message>
  <message name="helloResponse">
    <part name="parameters" element="tns:helloResponse"/>
  </message>
  <portType name="CalculatorWS">
    <operation name="sum">
      <input wsam:Action="http://calculator/CalculatorWS/sumRequest" message="tns:sum"/>
      <output wsam:Action="http://calculator/CalculatorWS/sumResponse" message="tns:sumResponse"/>
    </operation>
    <operation name="hello">
  
```

## WS in Java

[WSDL Mapping](#)

[Eccezioni e Fault](#)

[Contesto e ciclo di vita](#)

## Invocare un WS

[Un esempio riassuntivo](#)

## Supporto ai WS in Netbeans

[Il progetto per WS](#)

[Testing](#)

[WS Client](#)

## Conclusioni



# WS IN NETBEANS: TEST - 6

23: SOAP Web Services - 2

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 calculator.CalculatorWS.sum(int,int)
sum ( 3 , 5 )
```

```
public abstract java.lang.String calculator.CalculatorWS.hello(java.lang.String)
hello ( Vittorio )
```

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

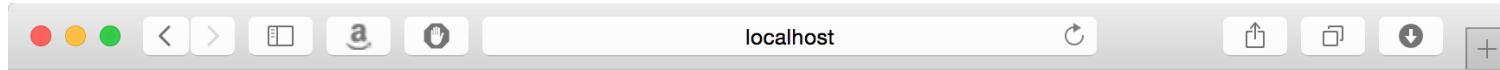
WS Client

Conclusioni



# WS IN NETBEANS: TEST - 7

23: SOAP Web Services - 2



## hello Method invocation

### Method parameter(s)

| Type             | Value    |
|------------------|----------|
| java.lang.String | Vittorio |

### Method returned

java.lang.String : "Hello Vittorio!"

### SOAP Request

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

### SOAP Response

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni



# PLAN

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni

23: SOAP Web Services - 2

### WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

### Invocare un WS

Un esempio riassuntivo

### Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

### Conclusioni



# WS IN NETBEANS: CLIENT - 1

23: SOAP Web Services - 2

New Project

Steps

1. Choose Project  
2. ...

Choose Project

Categories:

- Java
- JavaFX
- Java Web
- Java EE
- HTML5
- Maven
- NetBeans Modules
- Samples

Projects:

- Java Application
- Java Class Library
- Java Project with Existing Sources
- Java Free-Form Project

Description:

Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use an IDE-generated Ant build script to build, run, and debug your project.

Help < Back Next > Finish Cancel



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

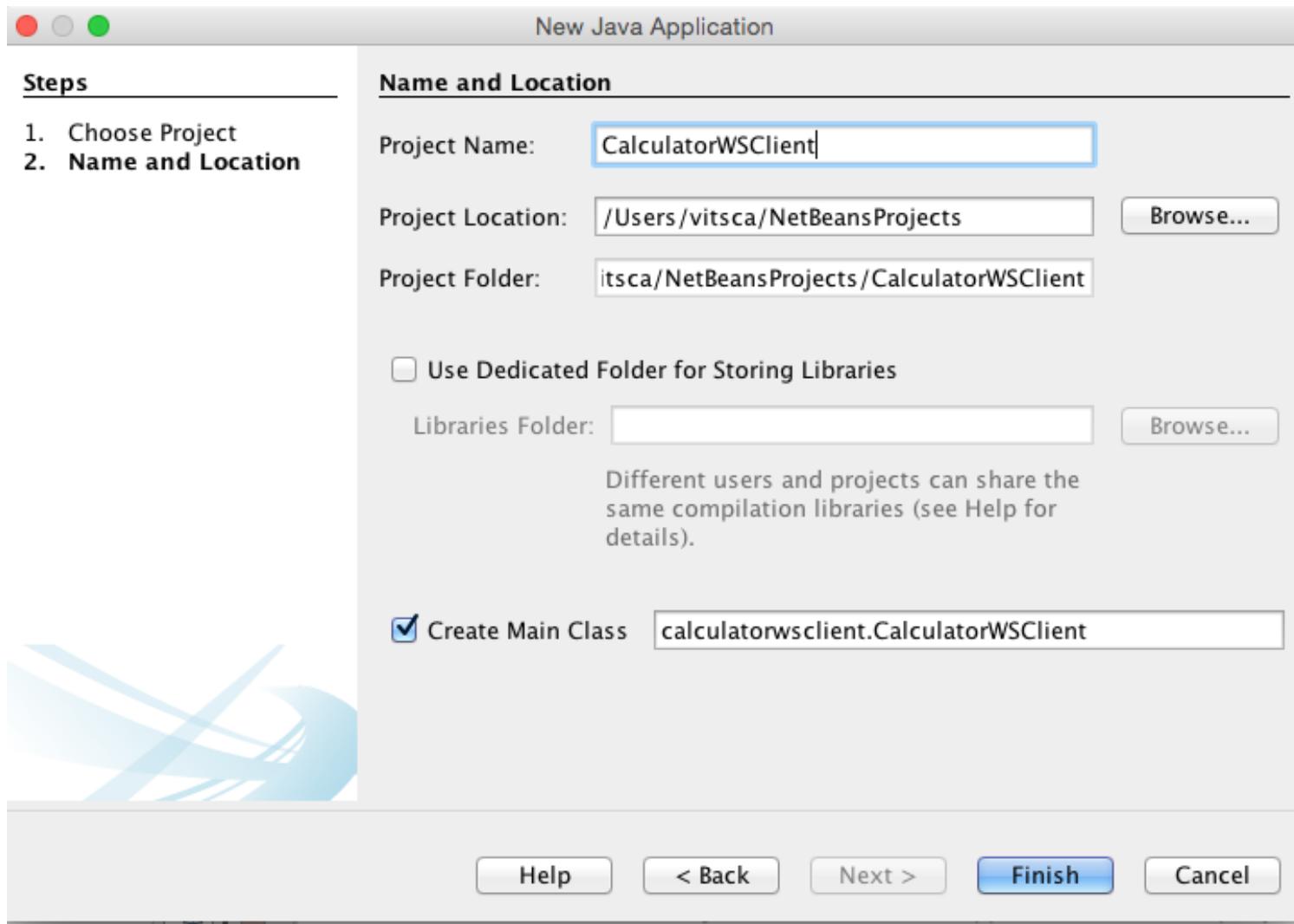
WS Client

Conclusioni



# WS IN NETBEANS: CLIENT - 2

23: SOAP Web Services - 2



## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

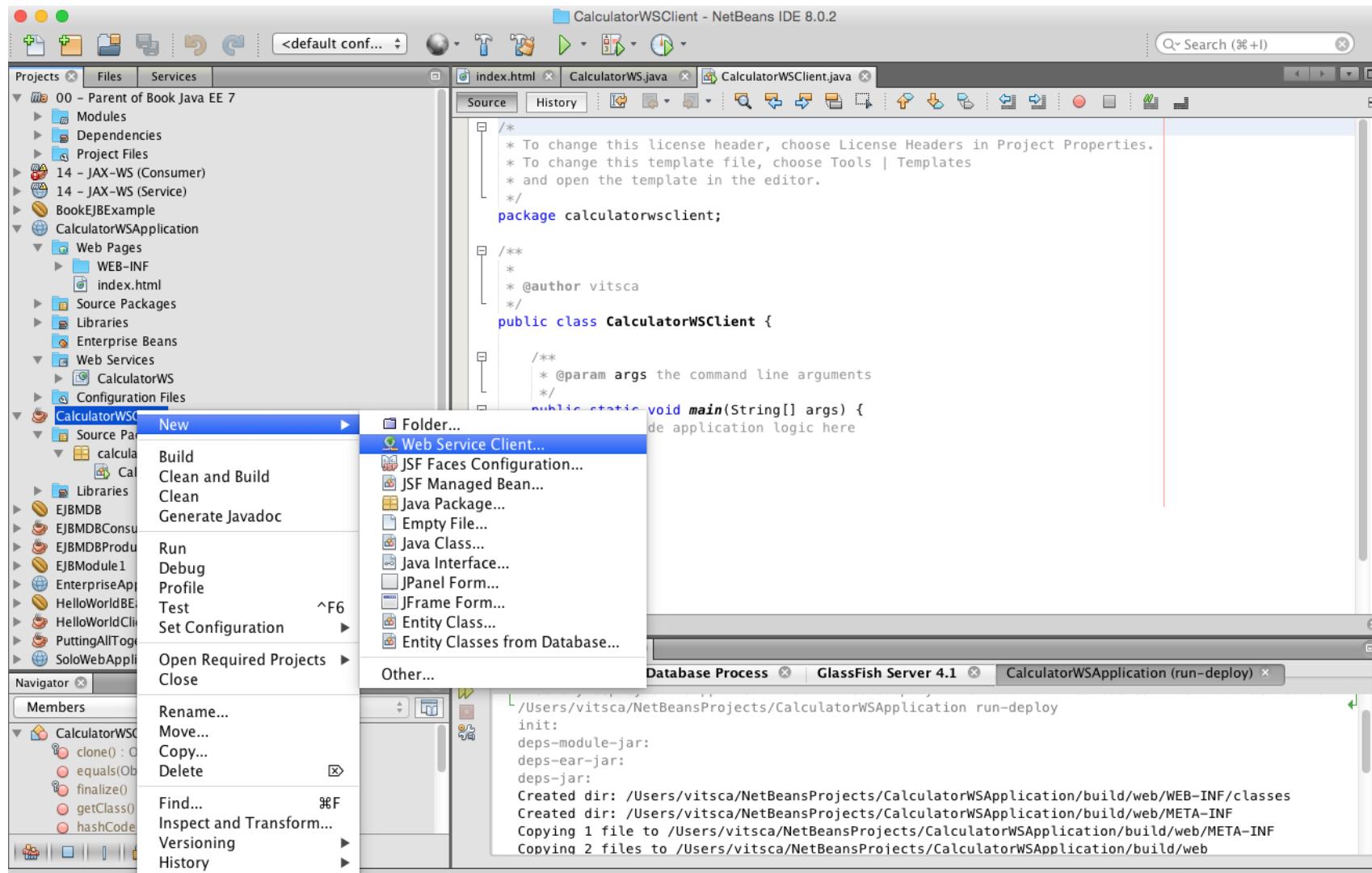
WS Client

## Conclusioni



# WS IN NETBEANS: CREIAMO UN WS CLIENT - 1

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

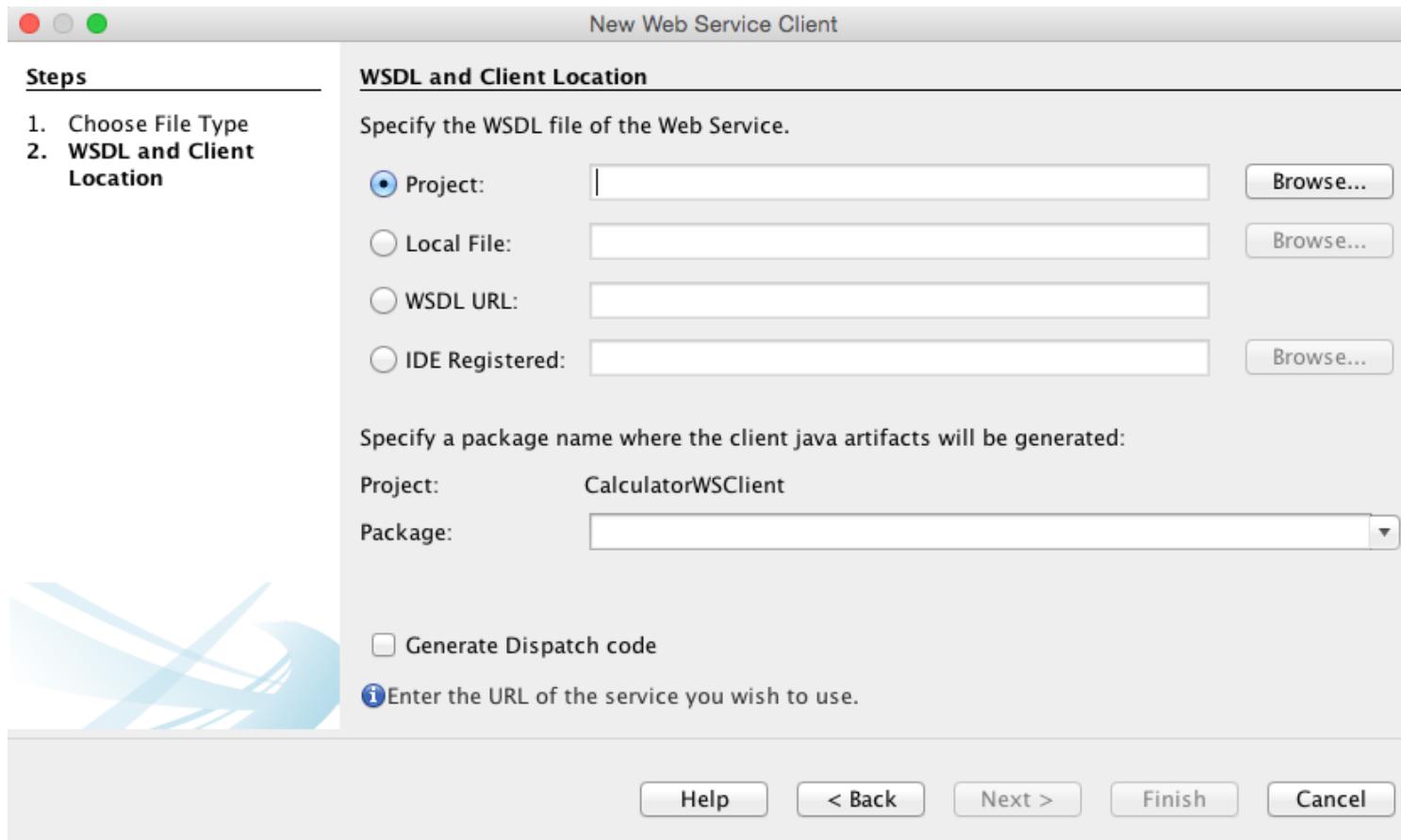
WS Client

Conclusioni



# WS IN NETBEANS: CREIAMO UN WS CLIENT - 2

23: SOAP Web Services - 2



## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

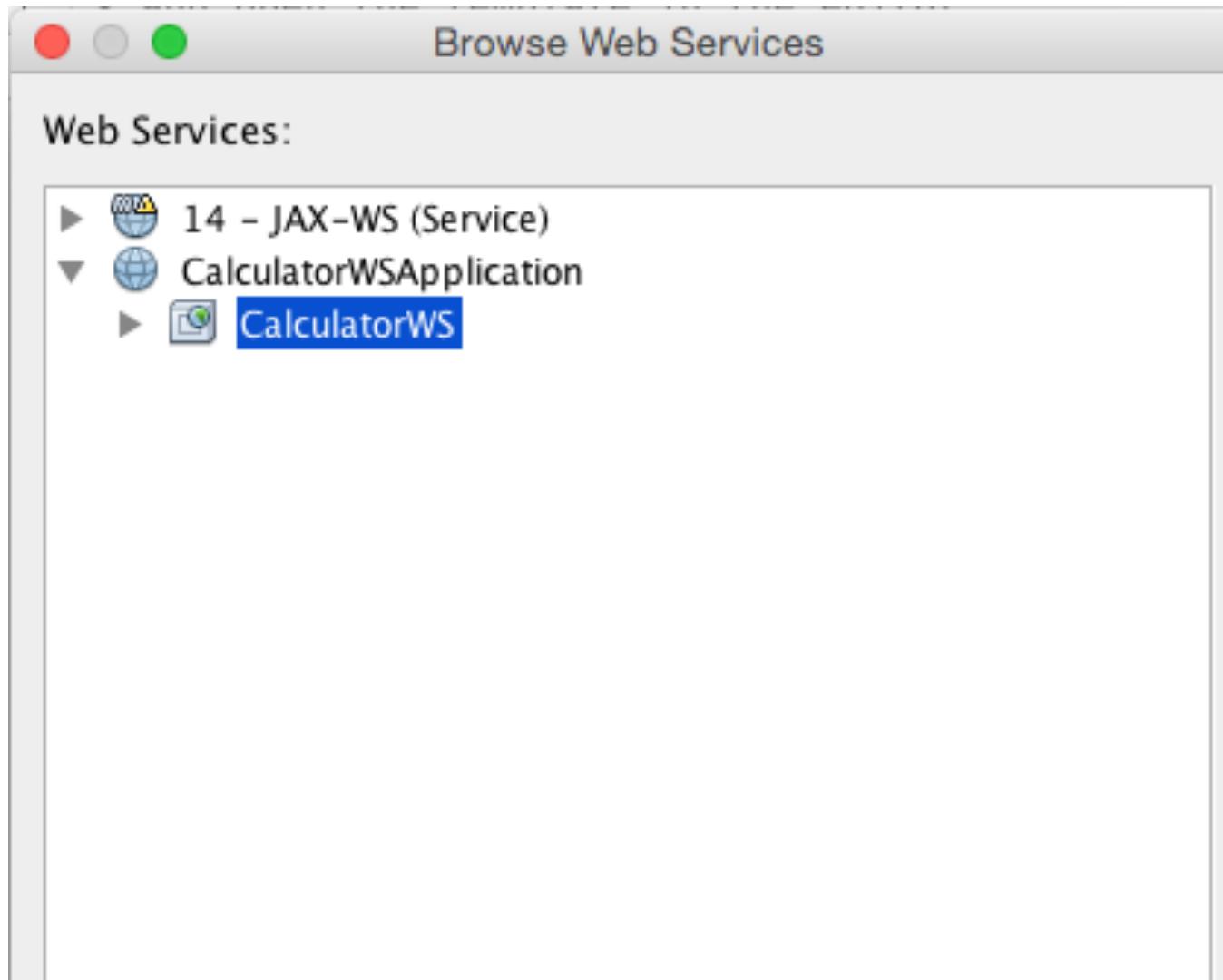
WS Client

## Conclusioni



# WS IN NETBEANS: CREIAMO UN WS CLIENT - 3

23: SOAP Web Services - 2



WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

Conclusioni

60/74



# WS IN NETBEANS: IL FILE WSDL

23: SOAP Web Services - 2

The screenshot shows the NetBeans IDE interface with the following details:

- Projects Tab:** Shows the project structure for "CalculatorWSClient". It includes a "Source Packages" folder containing "META-INF" (with "jax-ws-catalog.xml"), "META-INF.wsdl.localhost\_8080.CalculatorWS" (with "CalculatorWS.wsdl" and "CalculatorWS.xsd"), and "calculatorwsclient" (with "CalculatorWSClient.java"). It also contains a "Generated Sources (jax-ws)" folder with generated Java files like "CalculatorWS.java", "Hello.java", "HelloResponse.java", "ObjectFactory.java", "Sum.java", "SumResponse.java", and "package-info.java".
- CalculatorWS.wsdl Tab:** Displays the WSDL XML code for the service. The code defines two operations: "sum" and "hello". The "sum" operation has an input message "sum" and an output message "sumResponse". The "hello" operation has an input message "hello" and an output message "helloResponse". Both operations are associated with the port type "CalculatorWS".
- Output Tab:** Shows the build log for GlassFish Server 4.1. It lists the copied Java files: "calculator/Hello.java", "calculator/HelloResponse.java", "calculator/ObjectFactory.java", "calculator/Sum.java", "calculator/SumResponse.java", and "calculator/package-info.java". The log concludes with "BUILD SUCCESSFUL (total time: 1 second)".

WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

Invocare un WS

Un esempio riassuntivo

Supporto ai WS in Netbeans

Il progetto per WS

Testing

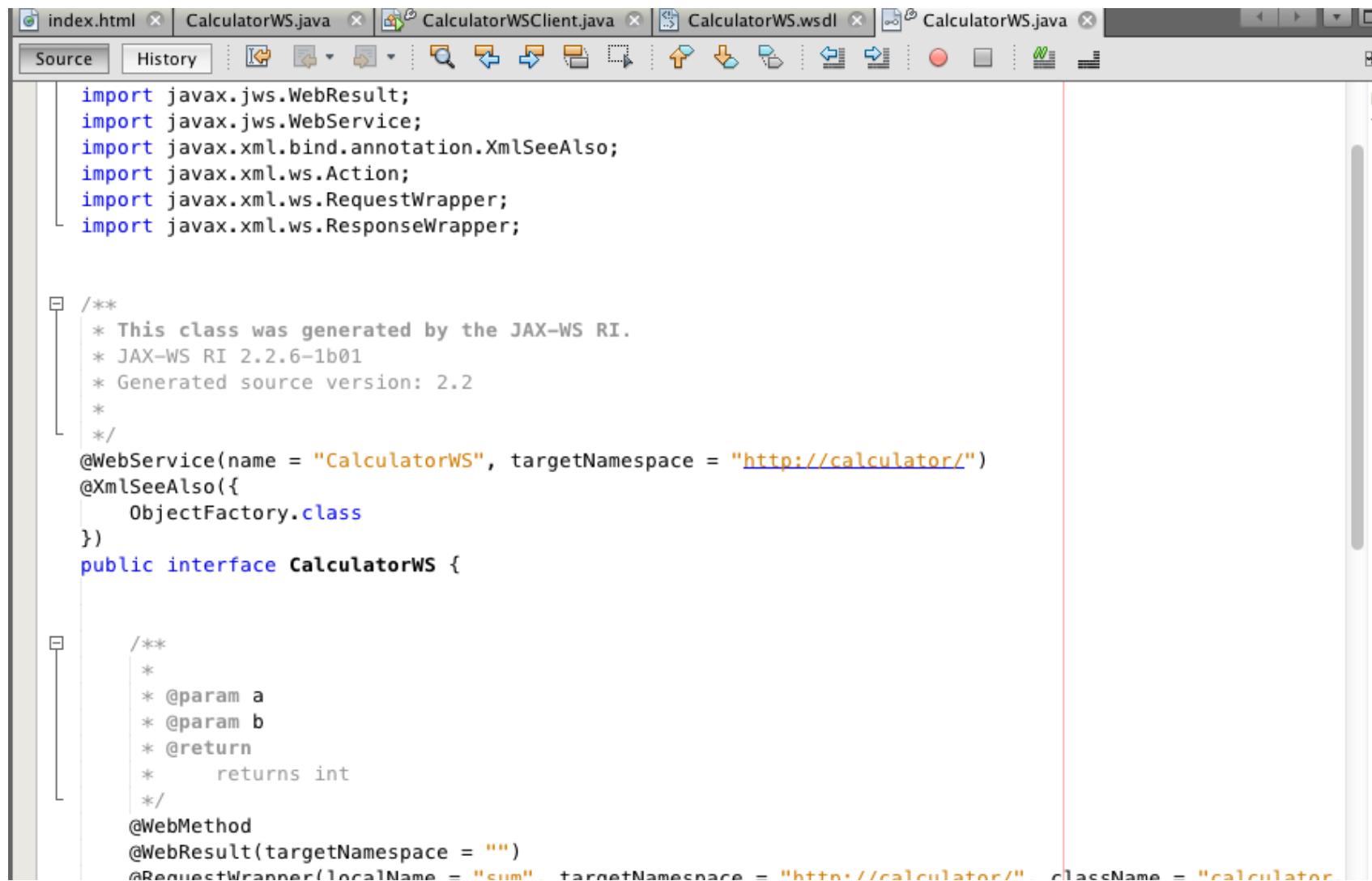
WS Client

Conclusioni



# WS IN NETBEANS: L'INTERFACCIA GENERATA - 1

23: SOAP Web Services - 2



The screenshot shows the NetBeans IDE interface with several tabs open at the top: index.html, CalculatorWS.java, CalculatorWSClient.java, CalculatorWS.wsdl, and another CalculatorWS.java tab. The main window displays the generated Java code for the `CalculatorWS` interface. The code includes imports for various JAX-WS annotations and classes, a class header with XML See Also information, and a public interface definition. A specific method is shown with its annotations: `@WebMethod`, `@WebResult`, `@RequestWrapper`, and `@ResponseWrapper`. The code is color-coded for syntax highlighting.

```
import javax.jws.WebResult;
import javax.jws.WebService;
import javax.xml.bind.annotation.XmlSeeAlso;
import javax.xml.ws.Action;
import javax.xml.ws.RequestWrapper;
import javax.xml.ws.ResponseWrapper;

/**
 * This class was generated by the JAX-WS RI.
 * JAX-WS RI 2.2.6-1b01
 * Generated source version: 2.2
 */
@WebService(name = "CalculatorWS", targetNamespace = "http://calculator/")
@XmlSeeAlso({
    ObjectFactory.class
})
public interface CalculatorWS {

    /**
     * 
     * @param a
     * @param b
     * @return
     *     returns int
     */
    @WebMethod
    @WebResult(targetNamespace = "")
    @RequestWrapper(localName = "sum", targetNamespace = "http://calculator/", className = "calculator")
    int sum(int a, int b);

}
```

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni

62/74



## WS IN NETBEANS: L'INTERFACCIA GENERATA - 2

## 23: SOAP Web Services - 2

The screenshot shows an IDE interface with multiple tabs at the top: index.html, CalculatorWS.java, CalculatorWSClient.java, CalculatorWS.wsdl, and CalculatorWS.java. The main area displays Java code for a web service client. The code includes annotations for web methods, request and response wrappers, and actions, along with XML schema imports.

```
/*
 * @param a
 * @param b
 * @return
 *     returns int
 */
@WebMethod
@WebResult(targetNamespace = "")
@RequestWrapper(localName = "sum", targetNamespace = "http://calculator/", className = "calculator.S")
@ResponseWrapper(localName = "sumResponse", targetNamespace = "http://calculator/", className = "cal")
@Action(input = "http://calculator/CalculatorWS/sumRequest", output = "http://calculator/CalculatorW
public int sum(
    @WebParam(name = "a", targetNamespace = "")
    int a,
    @WebParam(name = "b", targetNamespace = "")
    int b);

/*
 * @param name
 * @return
 *     returns java.lang.String
 */
@WebMethod
@WebResult(targetNamespace = "")
@RequestWrapper(localName = "hello", targetNamespace = "http://calculator/", className = "calculator
@ResponseWrapper(localName = "helloResponse", targetNamespace = "http://calculator/", className = "c
@Action(input = "http://calculator/CalculatorWS/helloRequest", output = "http://calculator/Calculato
public String hello(
```

WS in Java

WSDL Mapping

Eccezioni e Fault

## **Contesto e ciclo di vita**

## Invoke un WS

## Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

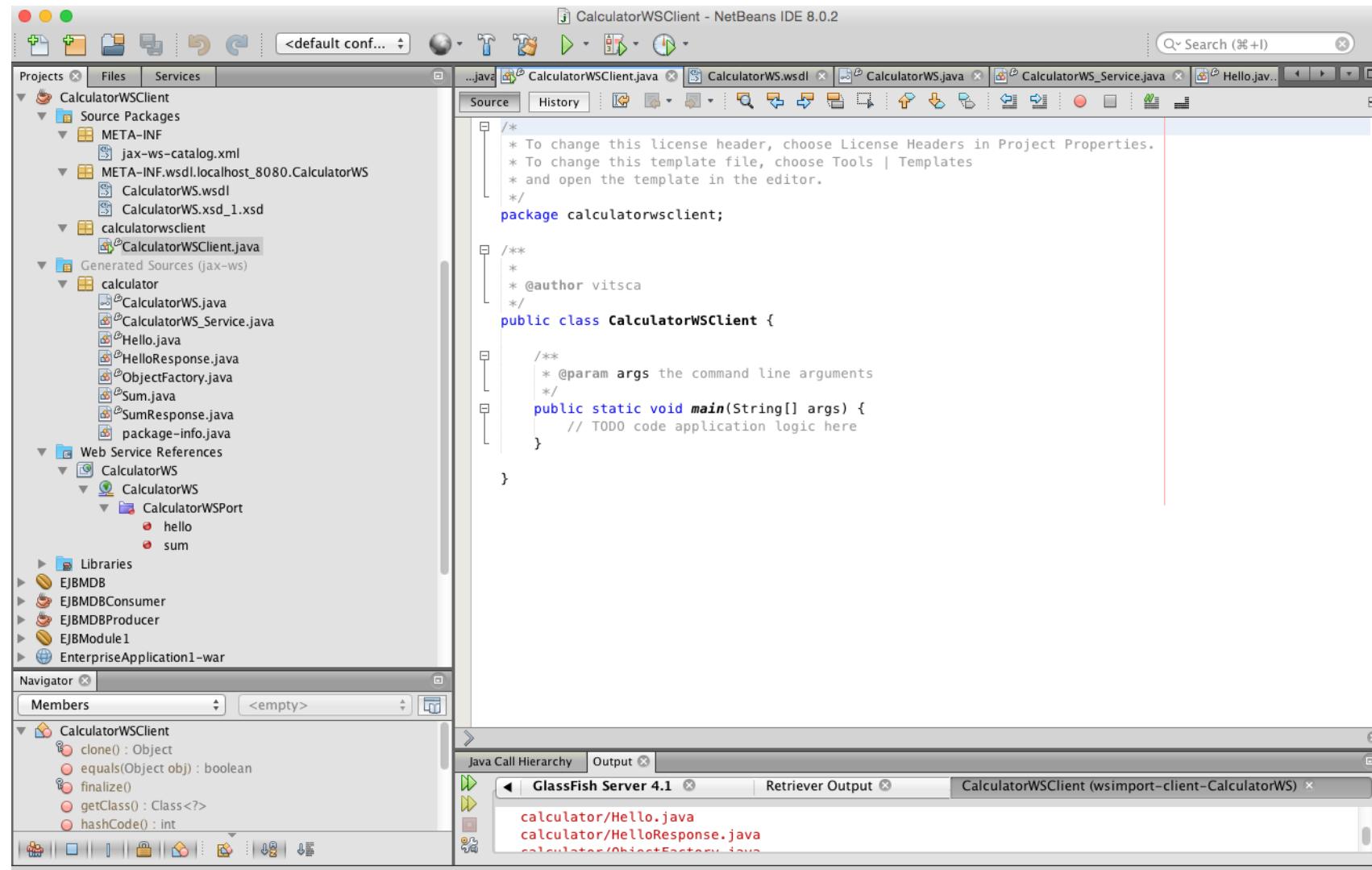
## Testing

WS Client

## Conclusioni



# WS IN NETBEANS: IL CLIENT (VUOTO)



## WS in Java

[WSDL Mapping](#)

[Eccezioni e Fault](#)

[Contesto e ciclo di vita](#)

## Invocare un WS

[Un esempio riassuntivo](#)

## Supporto ai WS in Netbeans

[Il progetto per WS](#)

[Testing](#)

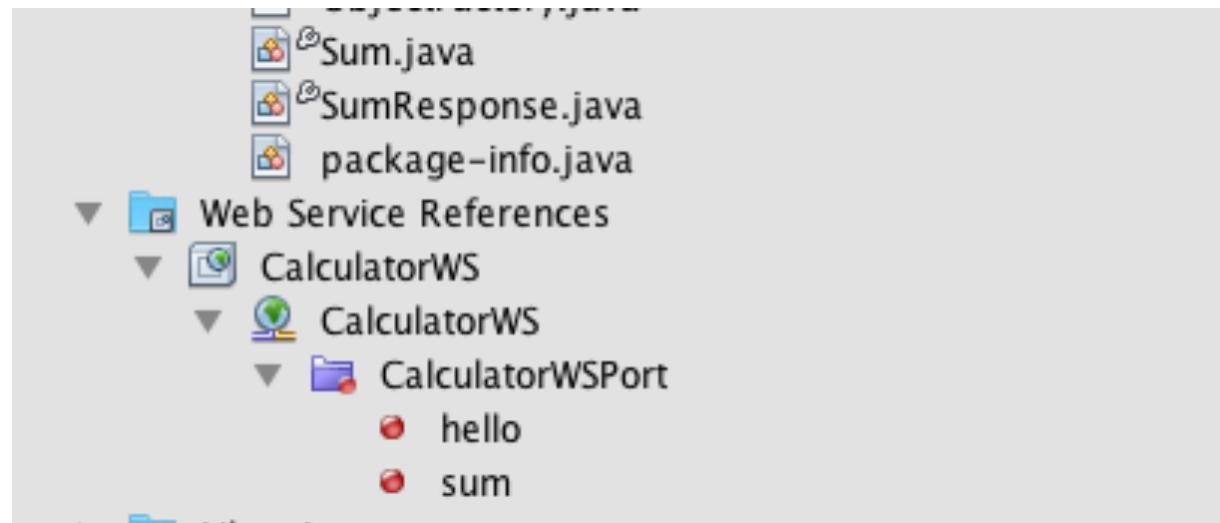
[WS Client](#)

## Conclusioni



# WS IN NETBEANS: DAL PROGETTO WS ...

23: SOAP Web Services - 2



## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# WS IN NETBEANS: DRAG&DROP OPERAZIONE hello

The screenshot shows the NetBeans IDE interface with the following details:

- Project Structure:** The top bar shows tabs for "...java", "CalculatorWSClient.java", "CalculatorWS.wsdl", "CalculatorWS.java", "CalculatorWS\_Service.java", and "Hello.j...".
- Toolbars:** Standard NetBeans toolbars for file operations, search, and navigation.
- Code Editor:** The main window displays the Java code for the `CalculatorWSClient` class. The code includes imports for `calculatorwsclient`, `calculator.CalculatorWS\_Service`, and `calculator.CalculatorWS`. It contains a `main` method that creates a service and port, and a `hello` static method that calls the `hello` operation on the port.

```
package calculatorwsclient;

/**
 * 
 * @author vitsca
 */
public class CalculatorWSClient {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
    }

    private static String hello(java.lang.String name) {
        calculator.CalculatorWS_Service service = new calculator.CalculatorWS_Service();
        calculator.CalculatorWS port = service.getCalculatorWSPort();
        return port.hello(name);
    }
}
```

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# WS IN NETBEANS: DRAG&DROP OPERAZIONE sum

```

package calculatorwsclient;

/*
 * @author vitsca
 */
public class CalculatorWSClient {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
    }

    private static String hello(java.lang.String name) {
        calculator.CalculatorWS_Service service = new calculator.CalculatorWS_Service();
        calculator.CalculatorWS port = service.getCalculatorWSPort();
        return port.hello(name);
    }

    private static int sum(int a, int b) {
        calculator.CalculatorWS_Service service = new calculator.CalculatorWS_Service();
        calculator.CalculatorWS port = service.getCalculatorWSPort();
        return port.sum(a, b);
    }
}

```

## WS in Java

[WSDL Mapping](#)

[Eccezioni e Fault](#)

[Contesto e ciclo di vita](#)

## Invocare un WS

[Un esempio riassuntivo](#)

## Supporto ai WS in Netbeans

[Il progetto per WS](#)

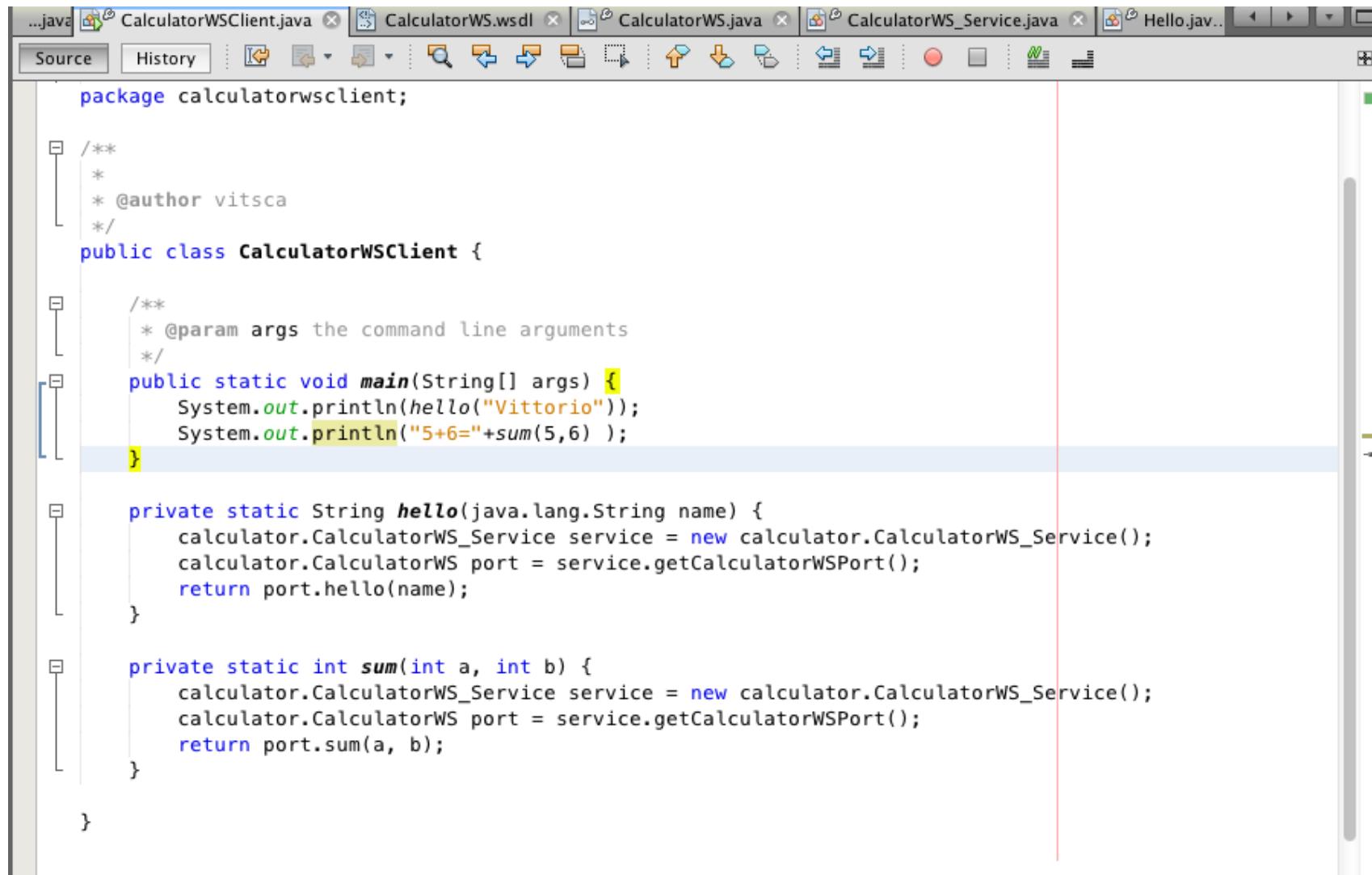
[Testing](#)

[WS Client](#)

## Conclusioni



# WS IN NETBEANS: ... UN “POCHINO” DI CODICE



The screenshot shows the NetBeans IDE interface with the following details:

- Project Explorer:** Shows files: ...java, CalculatorWSClient.java, CalculatorWS.wsdl, CalculatorWS.java, CalculatorWS\_Service.java, Hello.java.
- Toolbars:** Standard Java development toolbar.
- Code Editor:** Displays the `CalculatorWSClient.java` file content.

```

package calculatorwsclient;

/*
 * @author vitsca
 */
public class CalculatorWSClient {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        System.out.println(hello("Vittorio"));
        System.out.println("5+6=" + sum(5,6));
    }

    private static String hello(java.lang.String name) {
        calculator.CalculatorWS_Service service = new calculator.CalculatorWS_Service();
        calculator.CalculatorWS port = service.getCalculatorWSPort();
        return port.hello(name);
    }

    private static int sum(int a, int b) {
        calculator.CalculatorWS_Service service = new calculator.CalculatorWS_Service();
        calculator.CalculatorWS port = service.getCalculatorWSPort();
        return port.sum(a, b);
    }
}

```

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# WS IN NETBEANS: RUN ...

```

CalculatorWSClient - NetBeans IDE 8.0.2
CalculatorWSClient.java
CalculatorWS.wsdl
CalculatorWS.java
CalculatorWS_Service.java
Hello.java...
Source History ...
package calculatorwsclient;

/**
 * @author vitsca
 */
public class CalculatorWSClient {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        System.out.println(hello("Vittorio"));
        System.out.println("5+6=" + sum(5, 6));
    }

    private static String hello(java.lang.String name) {
        calculator.CalculatorWS_Service service = new calculator.CalculatorWS_Service();
        calculator.CalculatorWS port = service.getCalculatorWSPort();
        return port.hello(name);
    }

    private static int sum(int a, int b) {
        calculator.CalculatorWS_Service service = new calculator.CalculatorWS_Service();
        calculator.CalculatorWSClient port = new calculator.CalculatorWSClient();
        port.setPortName("CalculatorWSPort");
        port.hello("Vittorio");
        port.sum(5, 6);
    }
}

calculatorwsclient.CalculatorWSClient > main >
Java Call Hierarchy | Output ...
Java DB Database Process | GlassFish Server 4.1 | Retriever Output | CalculatorWSClient (run) ...
titles are up to date
wsimport-client-generate:
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/classes
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/empty
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/generated-sources/ap-source-out
put
Compiling 9 source files to /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/classes
Copying 3 files to /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/classes
compile:
run:
Hello Vittorio !
5+6=11
BUILD SUCCESSFUL (total time: 1 second)

```

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# WS IN NETBEANS: CON I RISULTATI...

```
Java Call Hierarchy | Output ×
Java DB Database Process × | GlassFish Server 4.1 × | Retriever Output × | CalculatorWSClient (run) ×
titles are up to date
wsimport-client-generate:
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/classes
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/empty
Created dir: /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/generated-sources/ap-source-out
put
Compiling 9 source files to /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/classes
Copying 3 files to /Users/vitsca/NetBeansProjects/CalculatorWSClient/build/classes
compile:
run:
Hello Vittorio !
5+6=11
BUILD SUCCESSFUL (total time: 1 second)
```

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni



# PLAN

23: SOAP Web Services - 2

## WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS

Testing

WS Client

## Conclusioni

### WS in Java

WSDL Mapping

Eccezioni e Fault

Contesto e ciclo di vita

### Invocare un WS

Un esempio riassuntivo

### Supporto ai WS in Netbeans

Il progetto per WS

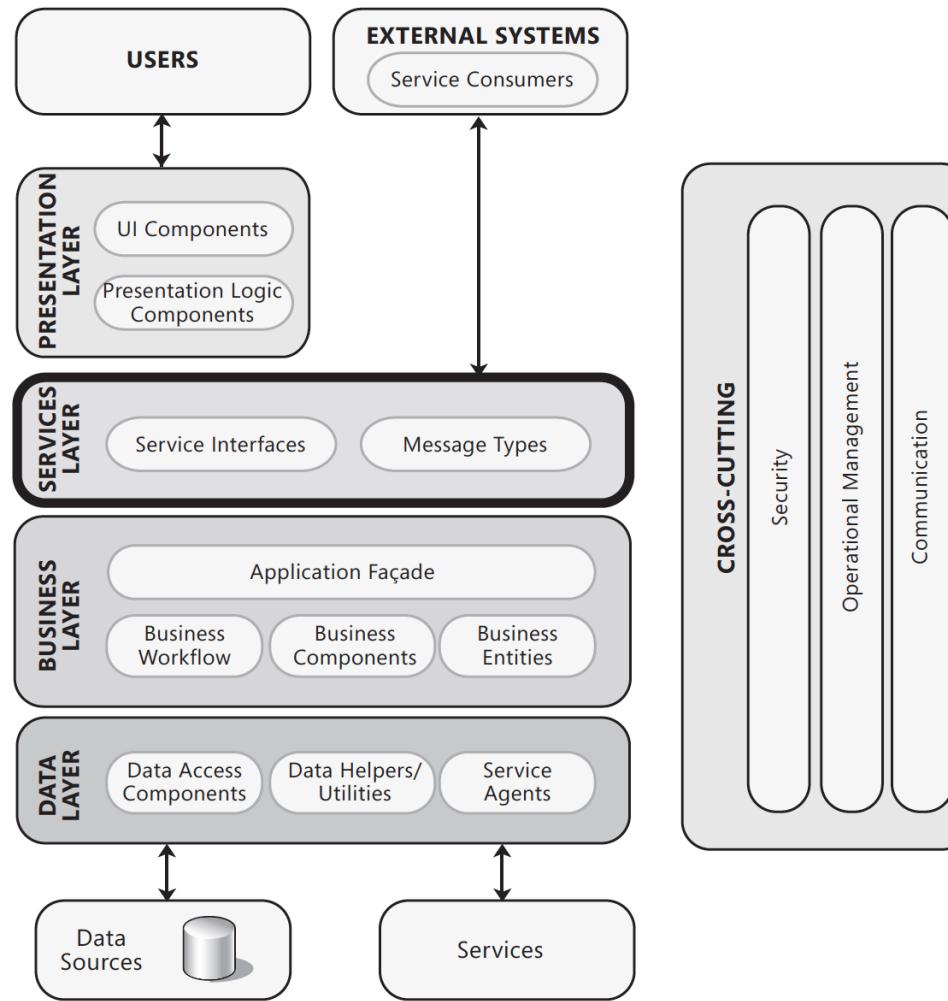
Testing

WS Client

### Conclusioni



# LA ARCHITETTURA CHE ABBIAMO VISTO



## WS in Java

WSDL Mapping  
Eccezioni e Fault  
Contesto e ciclo di vita

## Invocare un WS

Un esempio riassuntivo

## Supporto ai WS in Netbeans

Il progetto per WS  
Testing  
WS Client

## Conclusioni

