Feature Pack for Web Services Developing a JAXB application

What this exercise is about

The objective of this lab is to use the JAXB model to build a marshaller/demarshaller. The following is offered as a short refresher:

XML Schemas:

- An XML schema defines the allowable content and structure of an XML document
- An XML schema is similar to a Java class definition:
  - an XML document may declare that it conforms to a specific XML schema
  - a Java object conforms to a specific Java class
- XML schemas have largely replaced an earlier mechanism known as DTD (Document Type Definition)
- XML schemas are typically stored in files with “.xsd” extension
  - There are tools that will create an XML schema from an XML document

Bindings:

- A binding defines a correspondence between an XML schema and a collection of Java classes

    - Use this correspondence to map valid XML documents to and from instances of these classes

- Programmer works with Java classes/objects instead of having to deal with complexities of XML

- Bindings are typically created with a tool called a “binding compiler”

Marshalling:

- Marshalling is the process of turning one or more Java objects into an XML document

- Also called serializing

Unmarshalling:

- In the unmarshalling process you begin with a valid XML stream and convert it to Java objects

    - These objects are instances of the classes that were created by the schema compiler

- Also called deserializing

Lab requirements

The system on which you are installing WebSphere Application Server must be running a supported distributed operating system. For more information on supported operating systems, see the WebSphere Application Server V6.1 information center. In addition, the system must have the following resources available:

- An installation of IBM Web Services Feature Pack for WebSphere Application Server V6.1

- An installation of IBM Application Server Toolkit V6.1

What you should be able to do

At the end of this lab you should be able to:

- Given an XML Schema, you should be able to use the IBM WebSphere Application Server Toolkit to create and demonstrate a marshaller and demarshaller.

Introduction

The Feature Pack for Web Services builds on Web Services technologies available in WebSphere Application Server V6.1. A few of the major enhancements introduced in this Feature Pack are:

- A new JAX-WS programming model (with support for JAX-B, annotations, SOAP 1.2 and more)

- Support for WS-Reliable Messaging and WS-Secure Conversation

- Improved administration of Web Services configuration through the use of Policy Sets

Exercise instructions

Instructions and subsequent documentation use symbolic references to directories which are listed as follows:

Reference Variable	Windows Location	Linux UNIX Location
<WAS_HOME>	C:\WebSphere\AppServer	Linux /opt/WebSphere/AppServer AIX /usr/WebSphere/AppServer
<TEMP>	C:\temp	/tmp
<hostname>	Host name or host address for the machine where the profiles are being created	Host name or host address for the machine where the profiles are being created

Note:

This Lab exercise assumes that the following software is already installed on your system. Use the documentation associated with the software to identify prerequisite software and hardware and for installation and verification instructions:

- IBM WebSphere Application Server Version 6.1.0.2
- IBM WebSphere Application Server Version 6.1 Feature Pack for Web Services, Beta Release
- IBM WebSphere Application Server Toolkit Version 6.1

Part 1: Create a Java project

This section will describe the process of creating a Java project – an environment that will be used to build a Web Service application using the IBM WebSphere Application Server Toolkit version 6.1 (AST). The AST comes with IBM WebSphere Application Server V6.1 and this release of AST has specific support for the IBM WebSphere Application Server V6.1 Feature Pack for Web Services (WSFP).

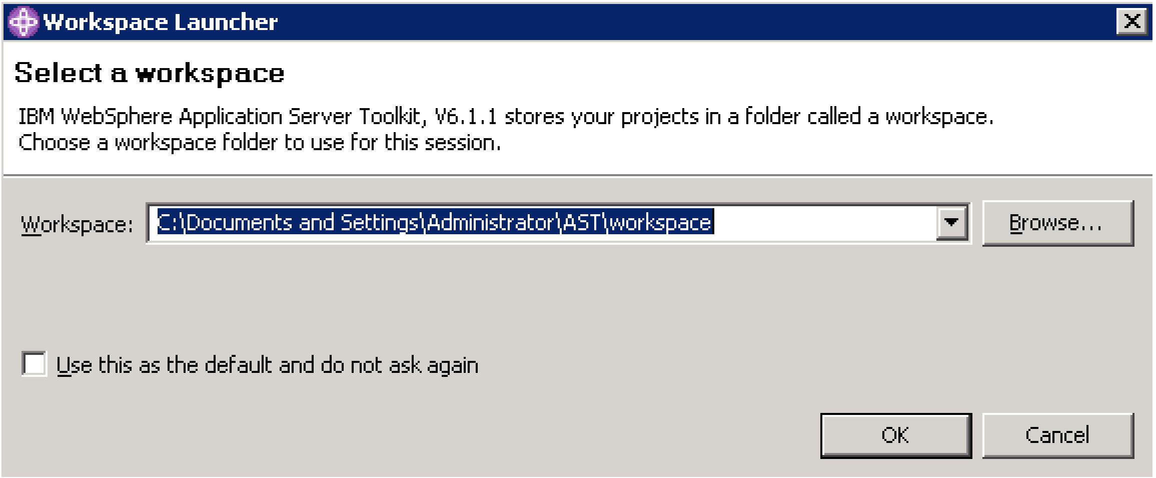
____ 1. Start the AST workspace.

To start the AST:

Using the Start Menu:

Start-> All Programs -> IBM WebSphere -> Application Server Toolkit V6.1-> Application Server Toolkit

____ 2. When prompted to select a workspace, keep the default workspace of **C:\Documents and Settings\Administrator\AST\workspace** and click **OK**.



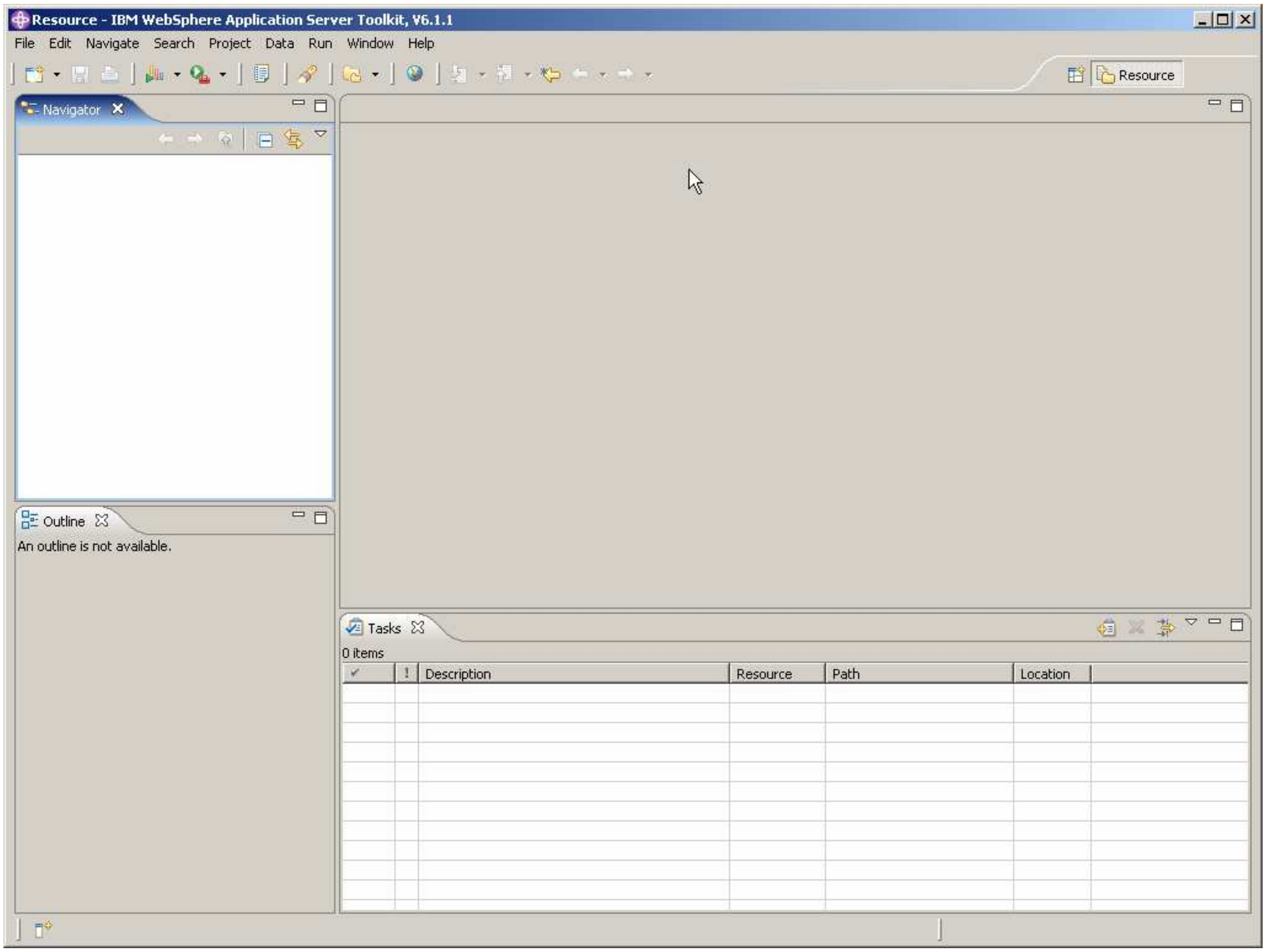
Select a workspace

IBM WebSphere Application Server Toolkit, V6.1.1 stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: C:\Documents and Settings\Administrator\AST\workspace

□ Use this as the default and do not ask again

____ 3. This will open the AST. Close the welcome screen by clicking the **X** on the welcome tab and you will see the screen shown below.


Resource - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Navigate Search Project Data Run Window Help
Resource
Navigator
Outline
Tasks

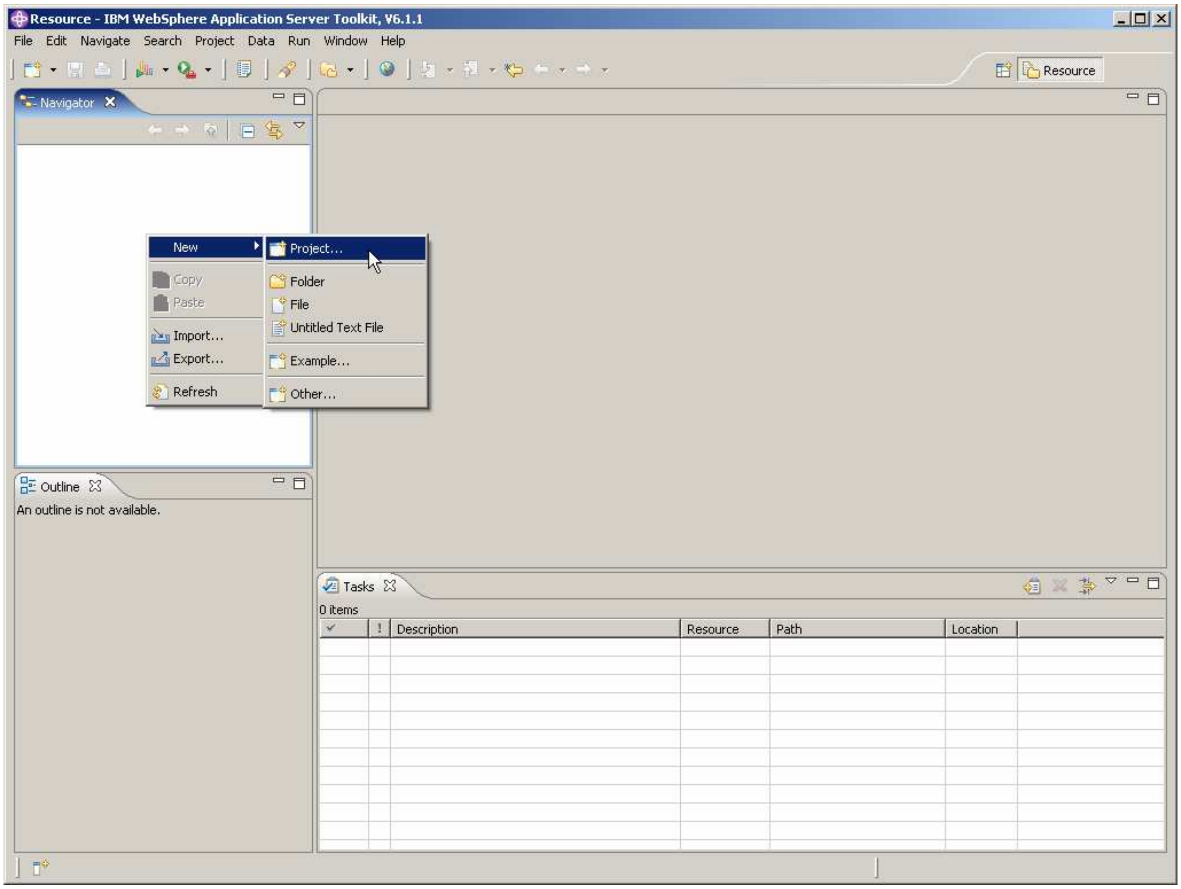
An outline is not available.

0 items

✓	!	Description	Resource	Path	Location

____ 4. Creating a new Java Application Project.

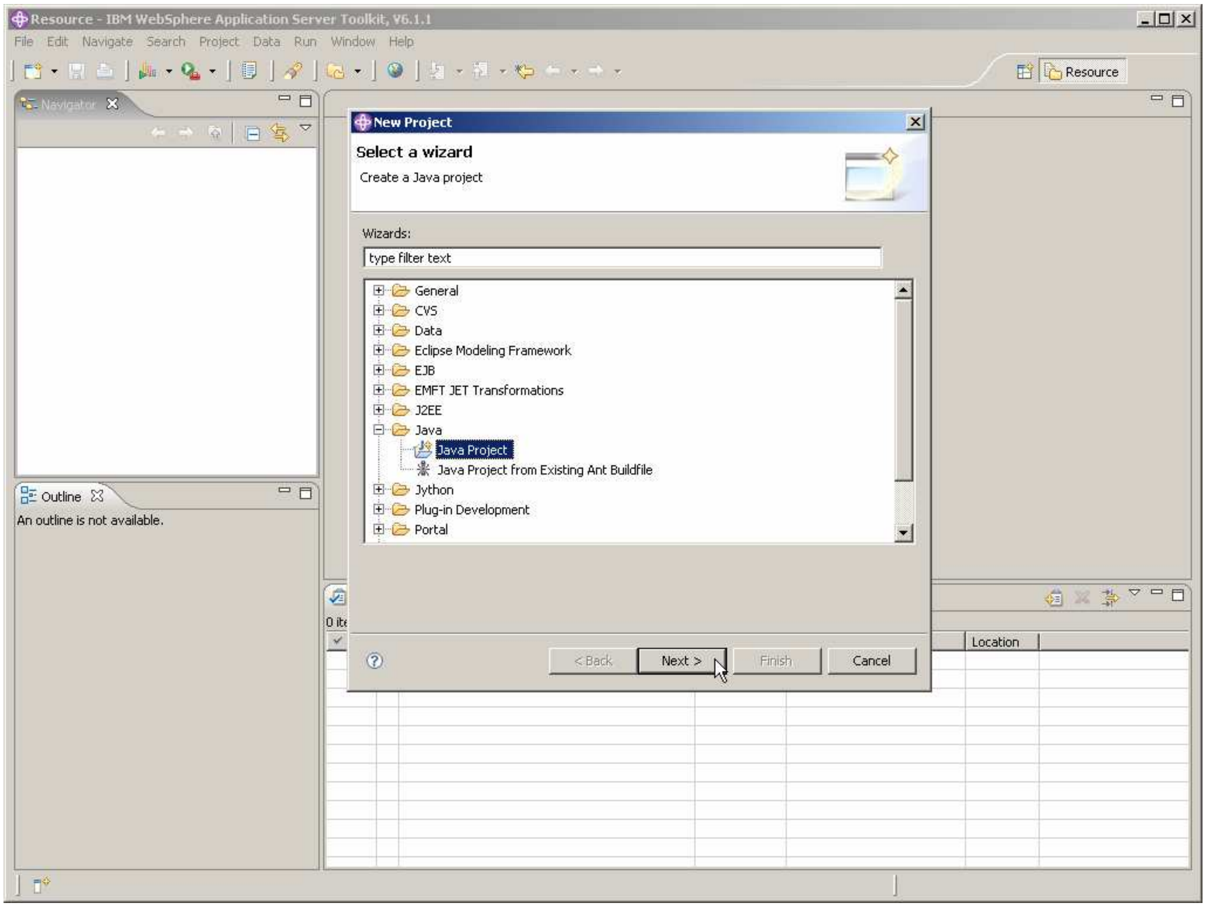
__ a. In the Project Explorer panel of the AST, right click in the empty space and select **New** then click on **Project.**



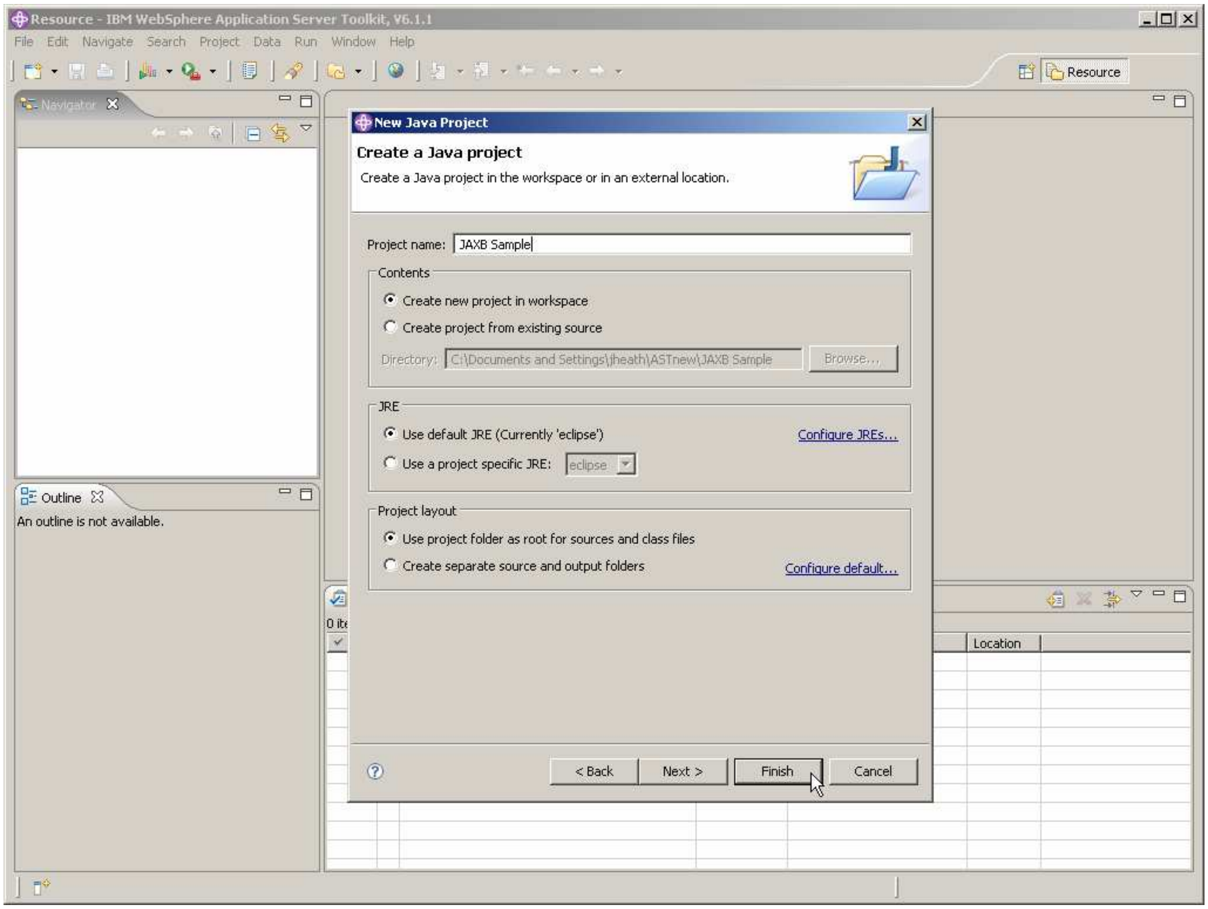
An outline is not available.

	!	Description	Resource	Path	Location

__ b. Expand the Java folder and select **Java Project.** Click **Next**.



__ c. For the Project Name enter **JAXB Sample**. Click **Finish**.



Resource - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Navigate Search Project Data Run Window Help
Resource
Navigator
Outline
Location

An outline is not available.

New Java Project

Create a Java project

Create a Java project in the workspace or in an external location.

Project name: JAXB Sample

Contents

● Create new project in workspace

○ Create project from existing source

Directory: C:\Documents and Settings\jheath\ASTnew\JAXB Sample Browse...

JRE

● Use default JRE (Currently 'eclipse') Configure JREs...

○ Use a project specific JRE: eclipse

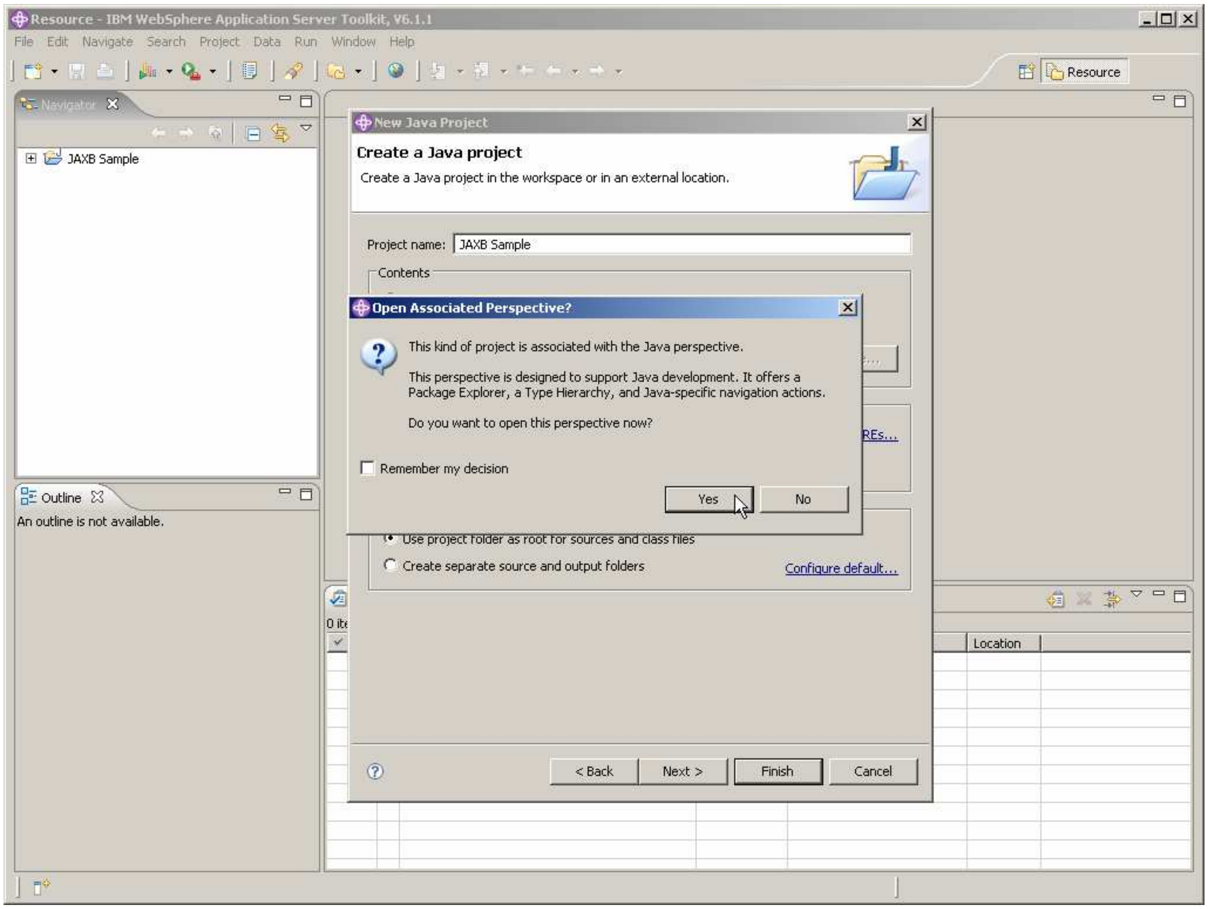
Project layout

● Use project folder as root for sources and class files

○ Create separate source and output folders Configure default...

< Back Next > Finish Cancel

____ 5. If you see a prompt asking to open the Java perspective, click **Yes**.



Resource - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Navigate Search Project Data Run Window Help
Resource
Navigator
JAXB Sample
Outline
Location
< Back
Next >
Finish
Cancel

New Java Project

Create a Java project

Create a Java project in the workspace or in an external location.

Project name: JAXB Sample

Contents

● Use project folder as root for sources and class files

○ Create separate source and output folders

Configure default...

Open Associated Perspective?

This kind of project is associated with the Java perspective.

This perspective is designed to support Java development. It offers a Package Explorer, a Type Hierarchy, and Java-specific navigation actions.

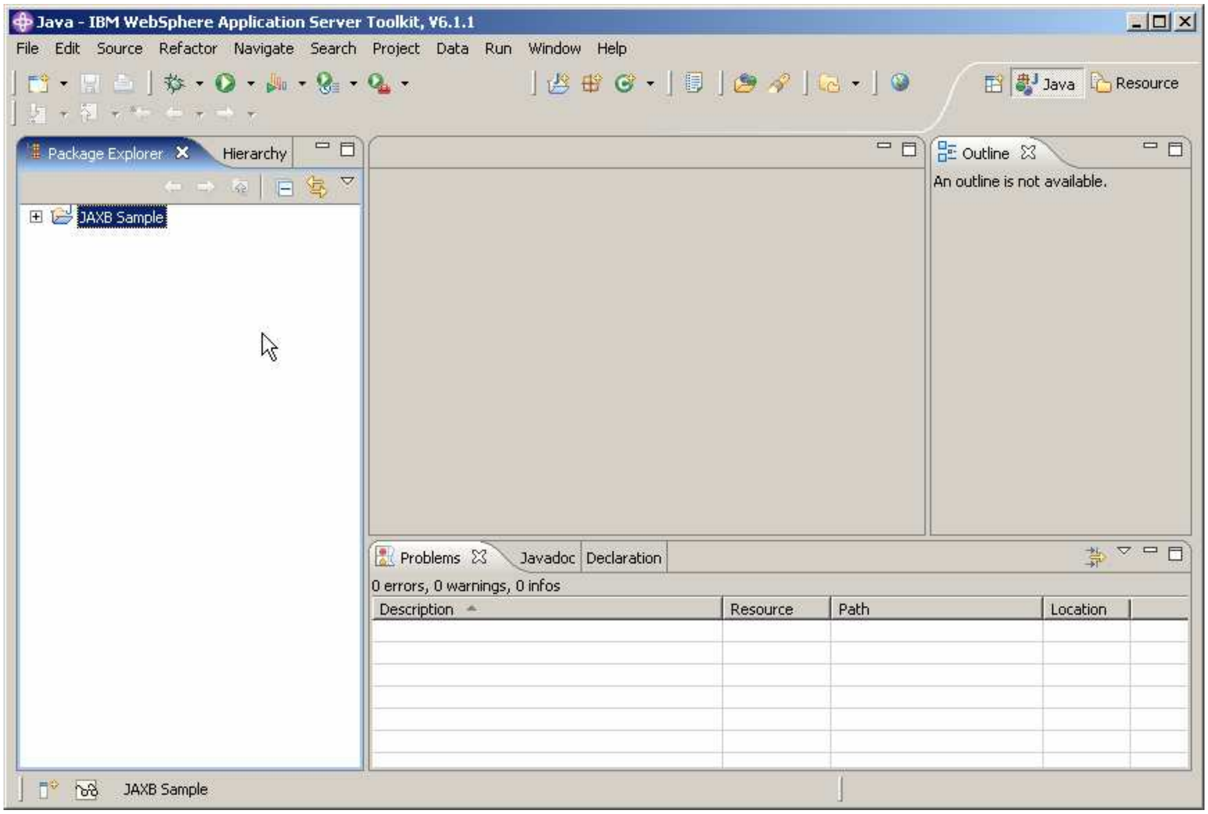
Do you want to open this perspective now?

☐ Remember my decision

Yes No

An outline is not available.

__ a. The Java Project should now be visible in the Project Explorer panel of the AST.


Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Hierarchy
JAXB Sample
Outline
Problems
Javadoc
Declaration
Description
Resource
Path
Location
JAXB Sample

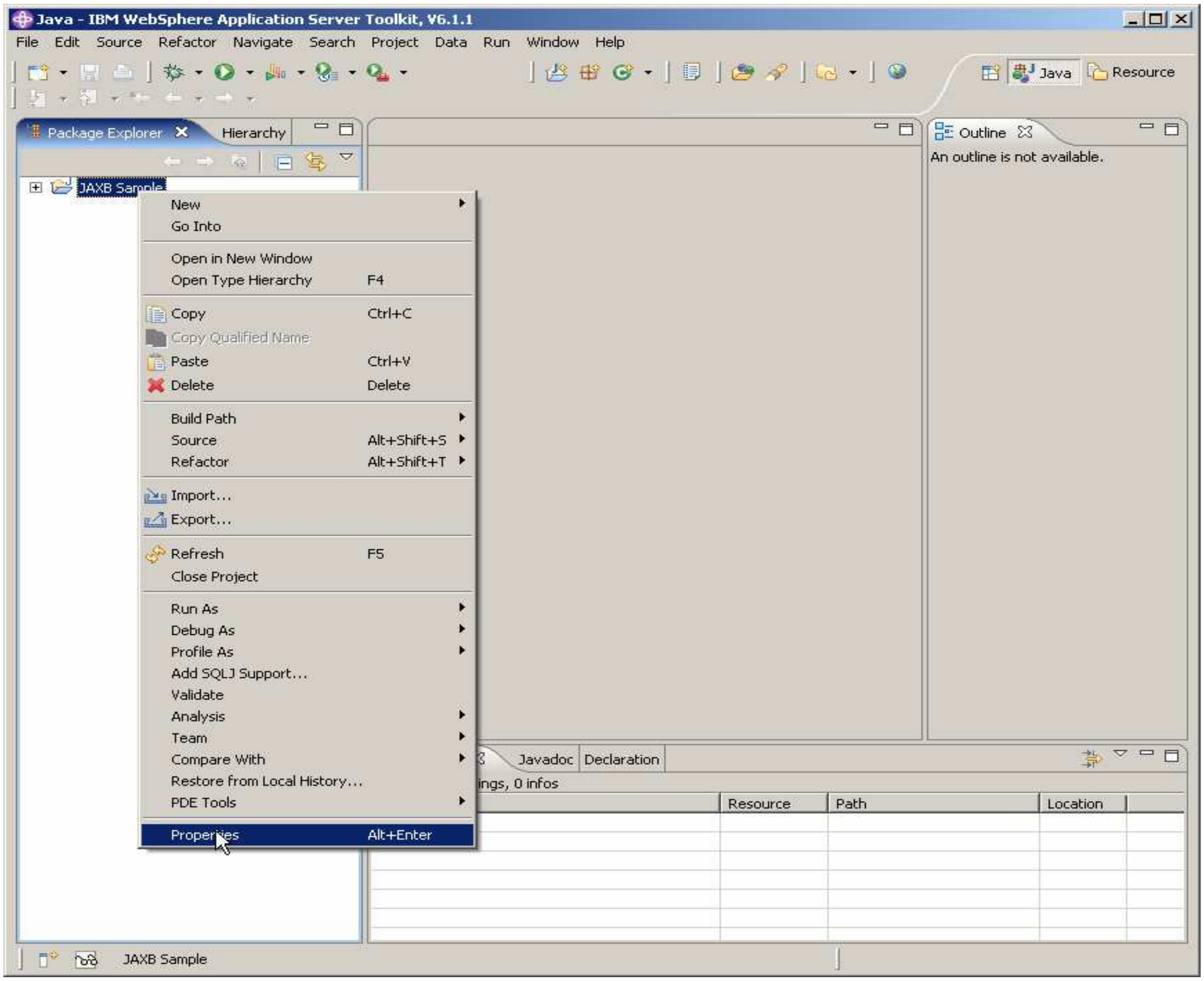
An outline is not available.

0 errors, 0 warnings, 0 infos

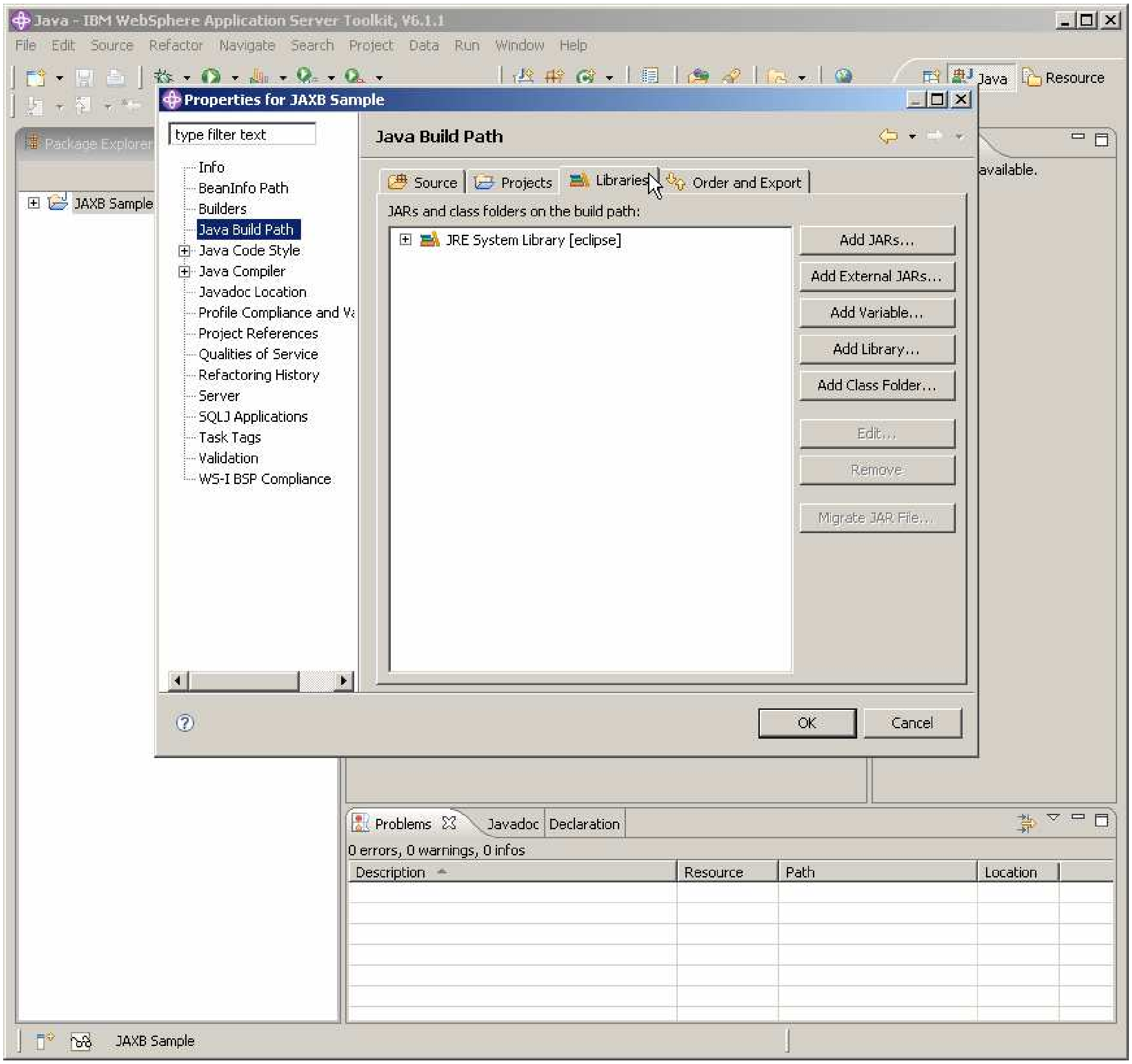
Part 2: Set some needed properties

In this section you will configure the new project.

____ 1. Right click on the **JAXB Sample** project and click on **Properties**.



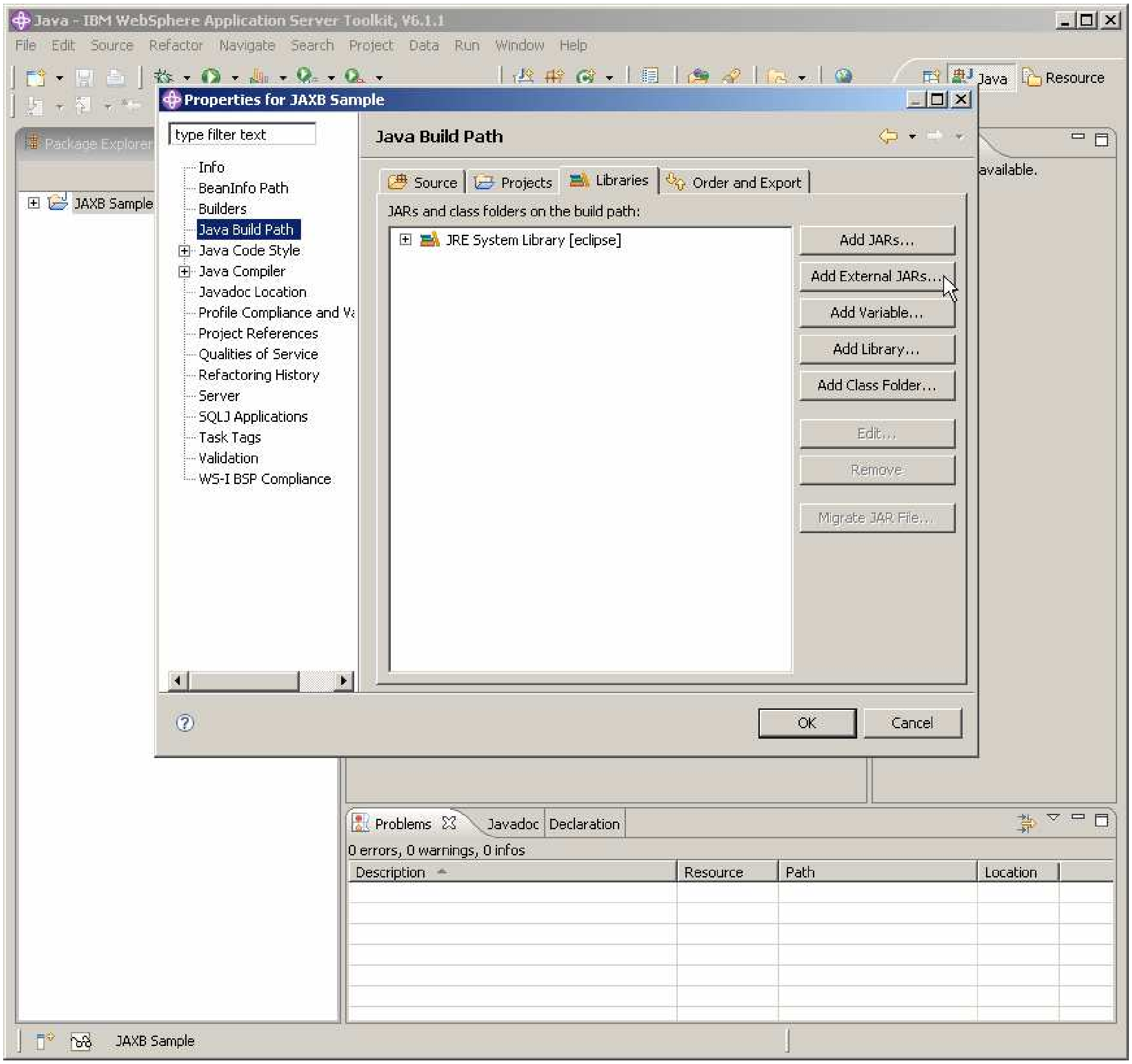
____ 2. Add to the build path by selecting **Java Build Path** then click the **Libraries** tab.



JARs and class folders on the build path:

0 errors, 0 warnings, 0 infos

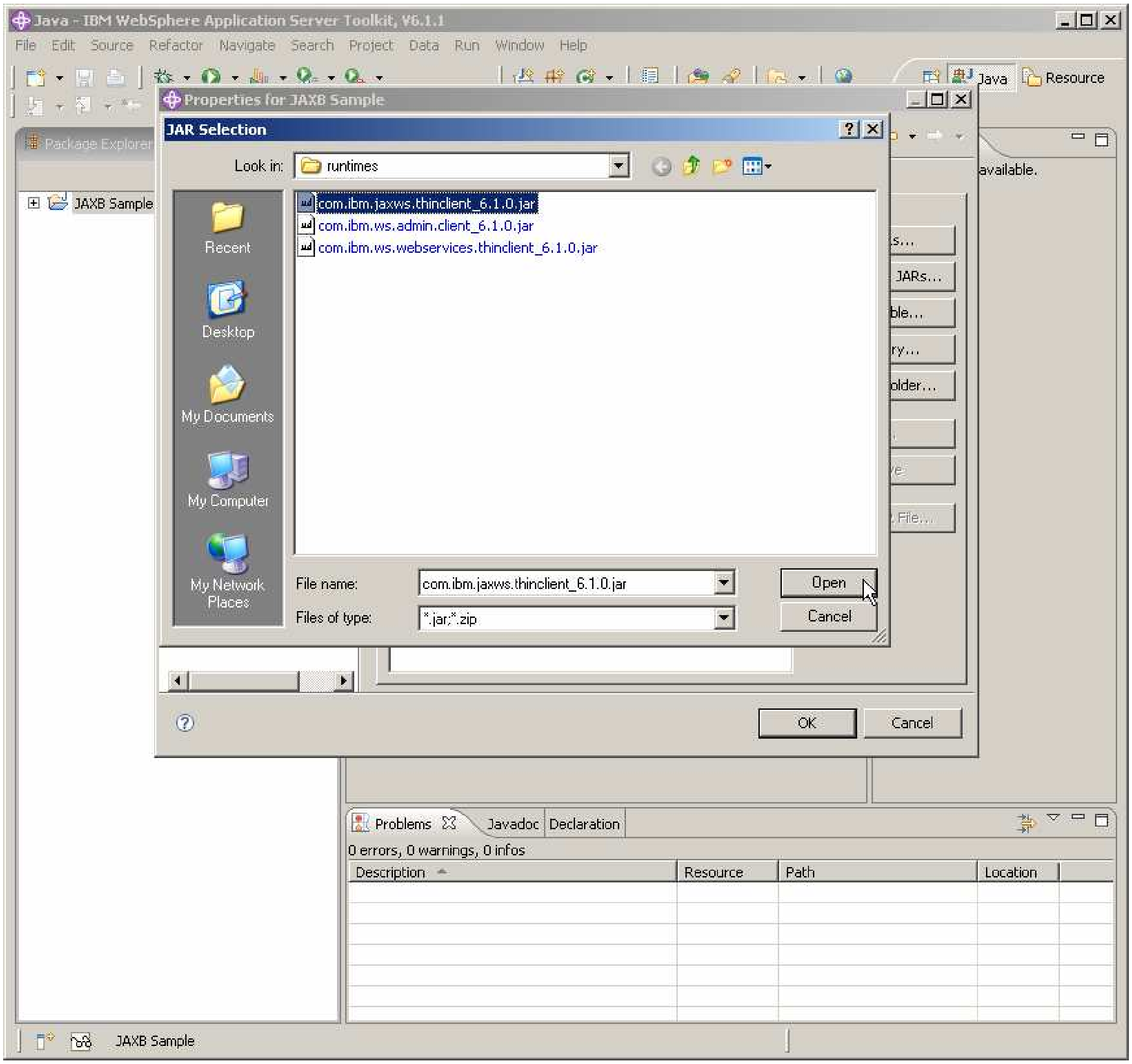
____ 3. Click on the button to **Add External Jars**.



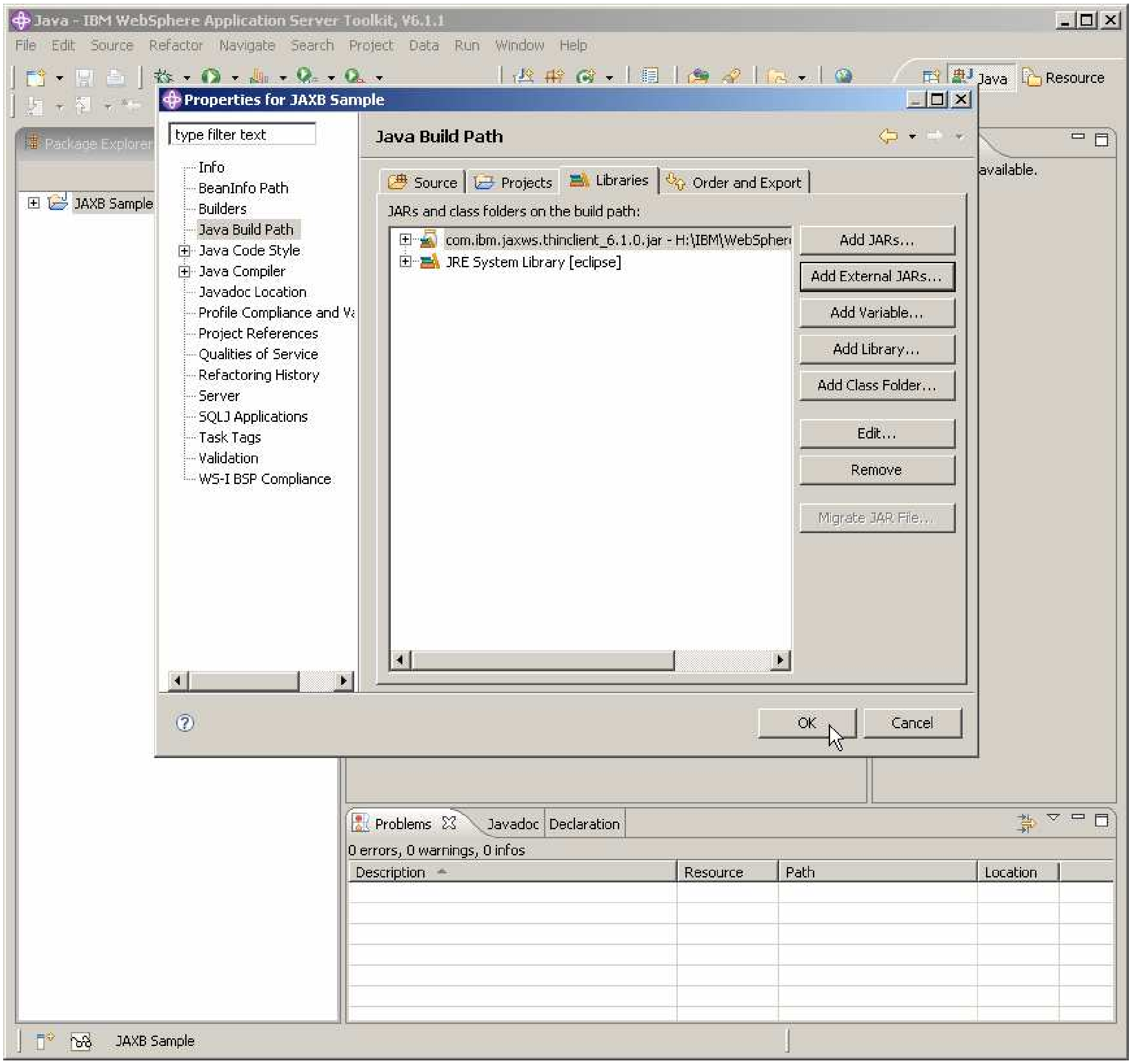
____ 4. Navigate to the **$WAS_HOME/runtimes** directory and select the file named:

com.ibm.jaxws.thinclient_6.1.0.jar

and click **Open**.


Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Properties for JAXB Sample
JAR Selection
Look in: runtimes
Recent
Desktop
My Documents
My Computer
My Network Places
com.ibm.jaxws.thinclient_6.1.0.jar
com.ibm.ws.admin.client_6.1.0.jar
com.ibm.ws.webservices.thinclient_6.1.0.jar
File name: com.ibm.jaxws.thinclient_6.1.0.jar
Files of type: *.jar;*.zip
Open
Cancel
OK
Cancel
Package Explorer
JAXB Sample
Problems
Javadoc
Declaration
0 errors, 0 warnings, 0 infos
Description
Resource
Path
Location
Java
Resource

____ 5. Click OK.


Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
JAXB Sample
Properties for JAXB Sample
type filter text
Info
BeanInfo Path
Builders
Java Build Path
Java Code Style
Java Compiler
Javadoc Location
Profile Compliance and V
Project References
Qualities of Service
Refactoring History
Server
SQLJ Applications
Task Tags
Validation
WS-I BSP Compliance
Java Build Path
Source
Projects
Libraries
Order and Export
JARs and class folders on the build path:
com.ibm.jaxws.thinclient_6.1.0.jar - H:\IBM\WebSpher
JRE System Library [eclipse]
Add JARs...
Add External JARs...
Add Variable...
Add Library...
Add Class Folder...
Edit...
Remove
Migrate JAR File...
OK
Cancel
available.
Problems
Javadoc
Declaration
0 errors, 0 warnings, 0 infos
Description
Resource
Path
Location
JAXB Sample

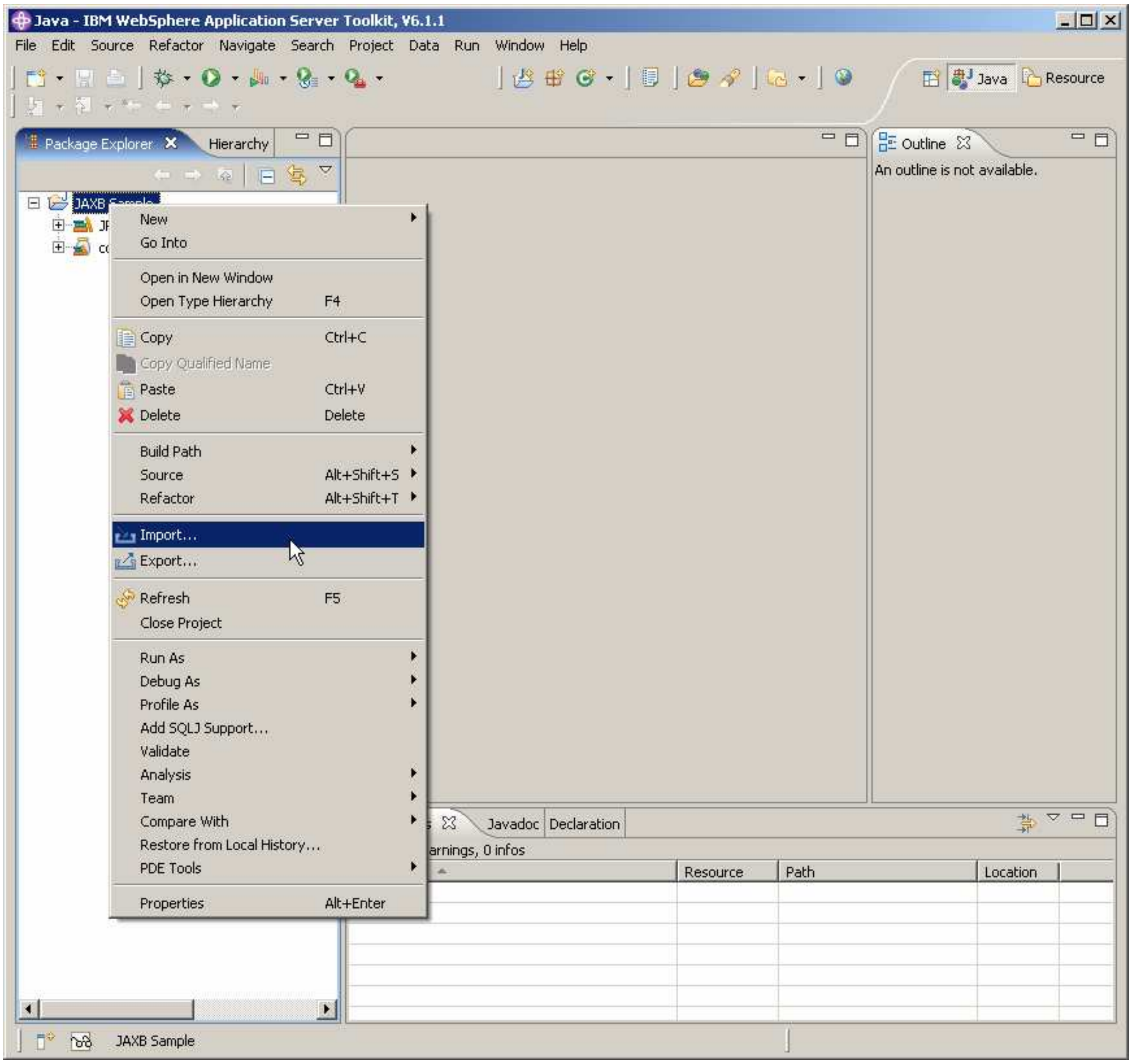
Part 3: Import the schema file

This section will describe the process of importing the schema file. XML schemas describe the data elements and relationships in an XML document. After a data mapping or binding exists, XML documents can be converted to and from Java objects. The schema being imported describes a simple library schema.

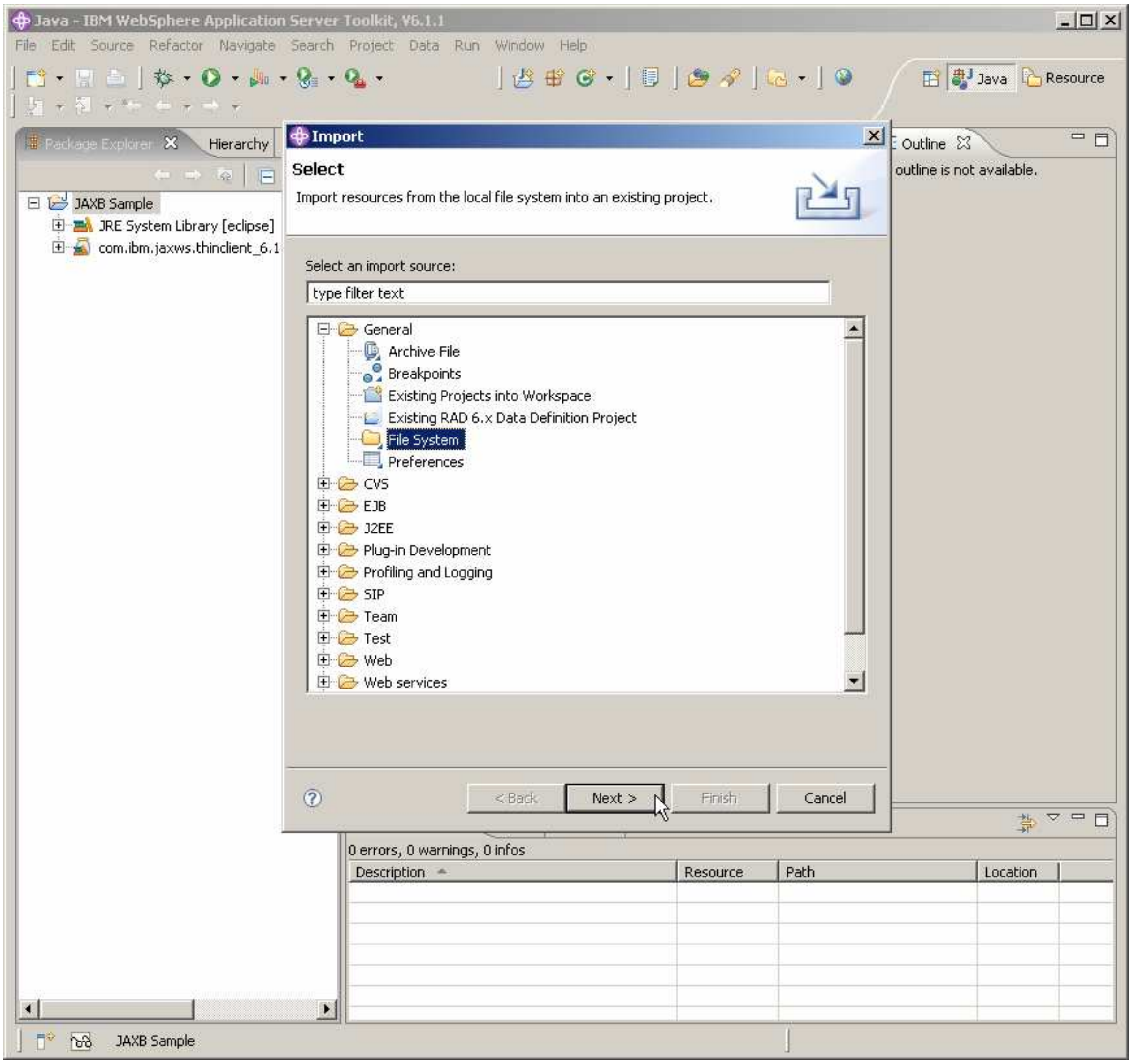
____ 1. Importing the schema

__ a. Expand the **JAXB Sample** Java Application project in the Project Explorer panel of the AST.

__ b. Click **Import…..**



____ 2. Expand **General**, select **File System** then click **Next**.


Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Hierarchy
JAXB Sample
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1
Outline
outline is not available.
Import

Select

Import resources from the local file system into an existing project.

Select an import source:

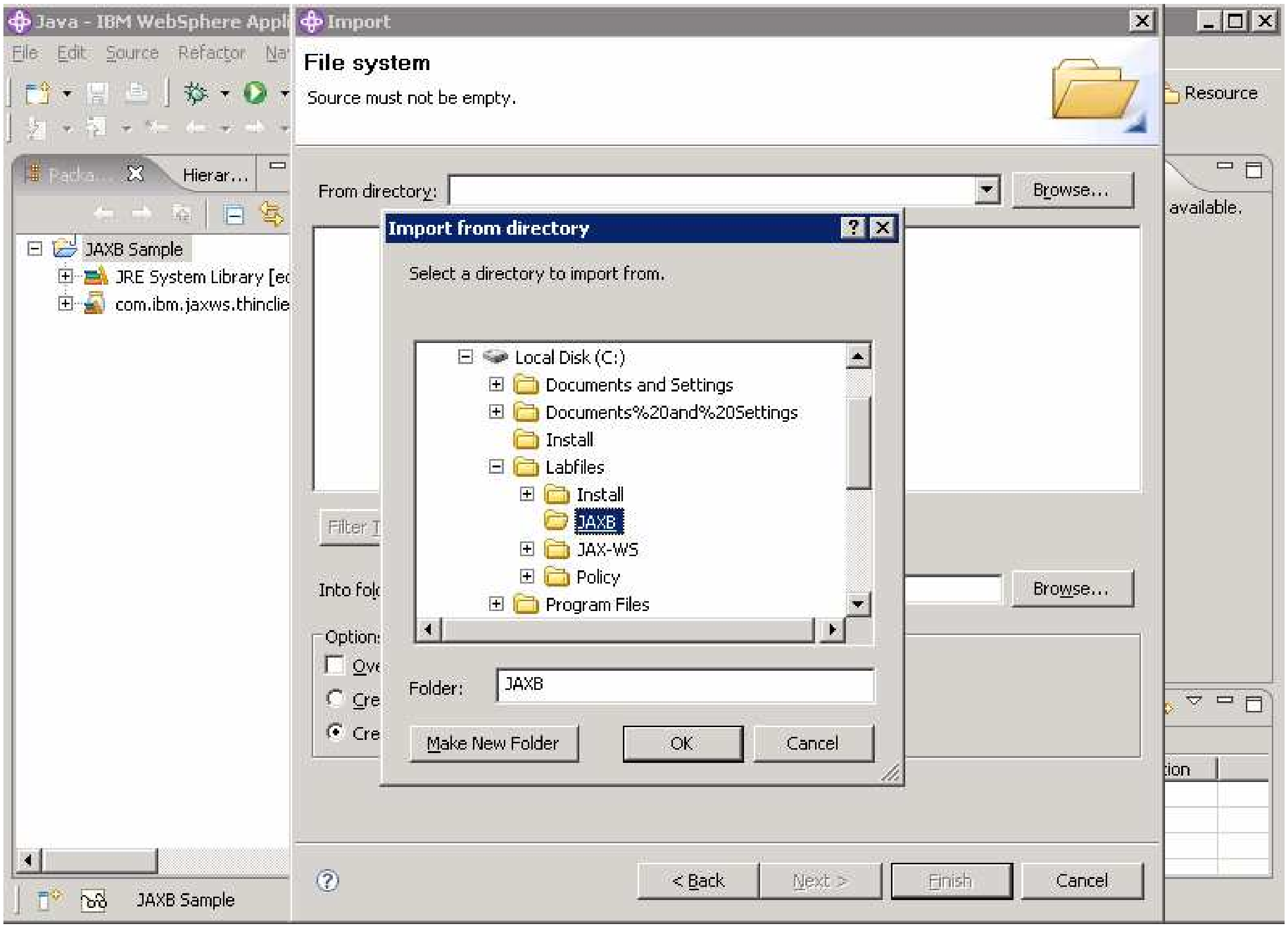
type filter text

General
Archive File
Breakpoints
Existing Projects into Workspace
Existing RAD 6.x Data Definition Project
File System
Preferences
CVS
EJB
J2EE
Plug-in Development
Profiling and Logging
SIP
Team
Test
Web
Web services
< Back
Next >
Finish
Cancel

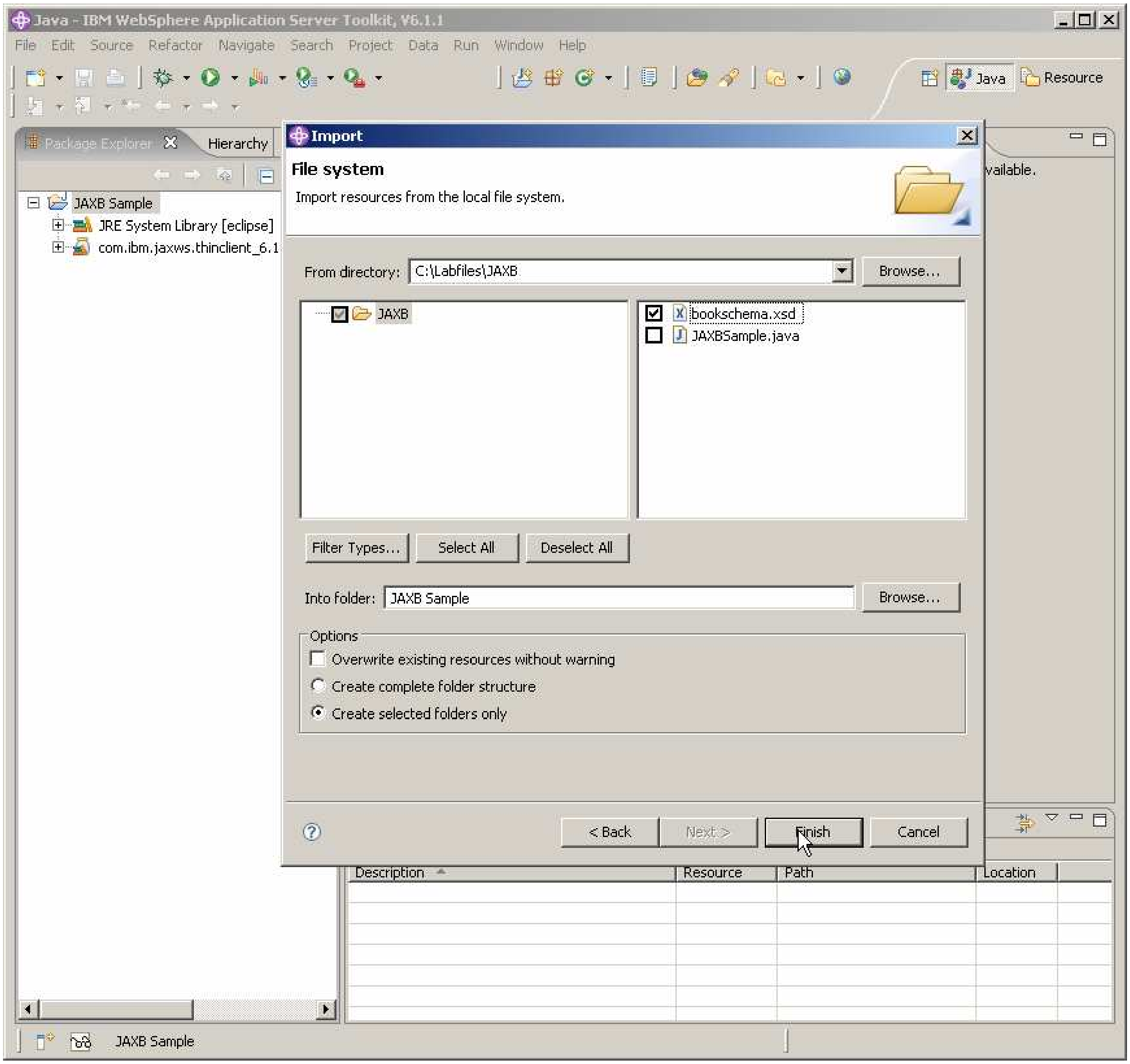
0 errors, 0 warnings, 0 infos

Description	Resource	Path	Location

__ a. Navigate to **C:\Labfiles\JAXB** and click **OK**.



____ 3. Check the box by the file named **bookschema.xsd** and click **Finish**.


Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Hierarchy
JAXB Sample
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1
vailable.
Description
Resource
Path
Location
JAXB Sample

Import

File system

Import resources from the local file system.

From directory: C:\Labfiles\JAXB

☑ JAXB

☑ bookschema.xsd

□ JAXBSample.java

Into folder: JAXB Sample

Options

□ Overwrite existing resources without warning

○ Create complete folder structure

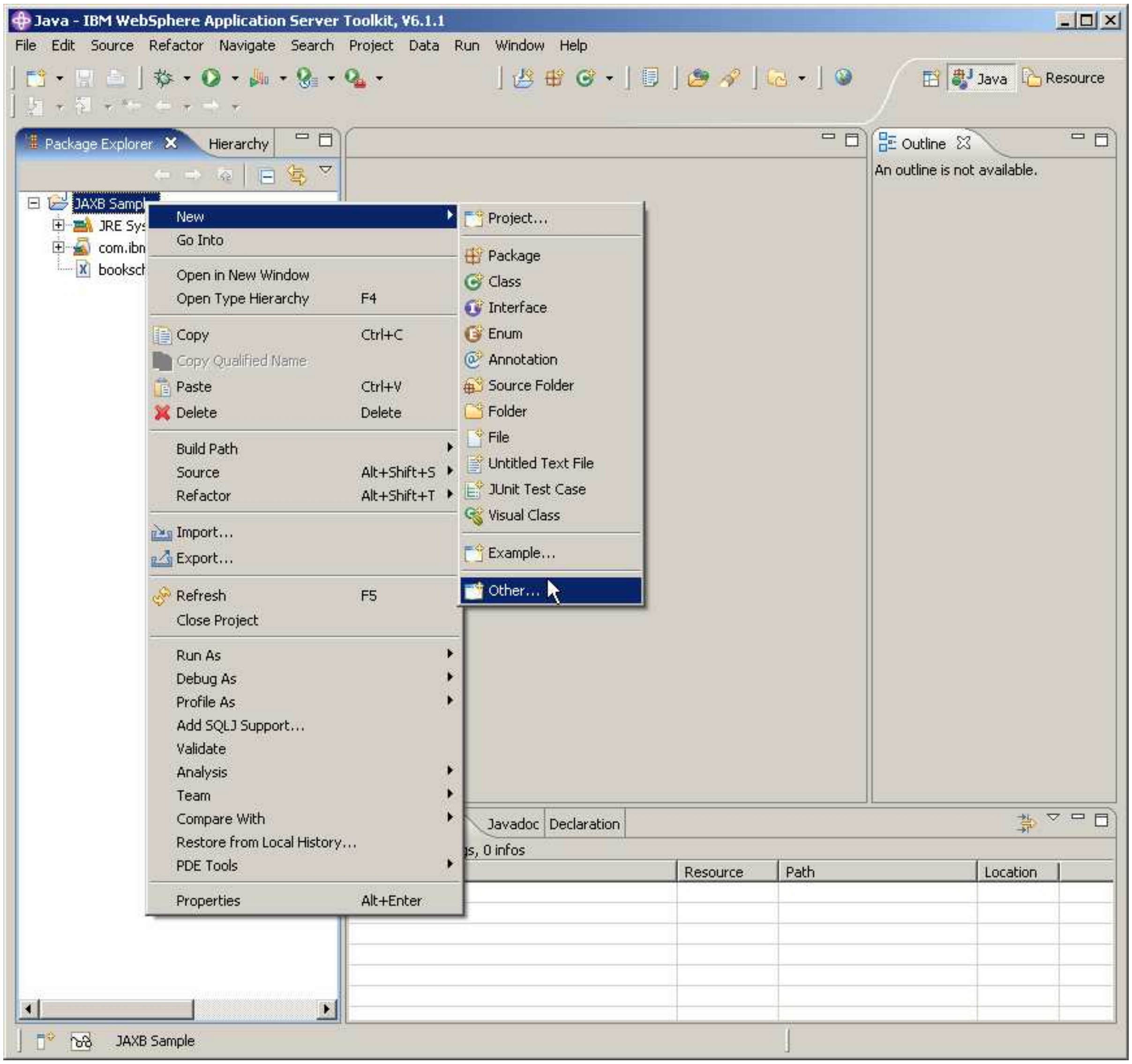
● Create selected folders only

Browse...
Filter Types...
Select All
Deselect All
Browse...
< Back
Next >
Finish
Cancel

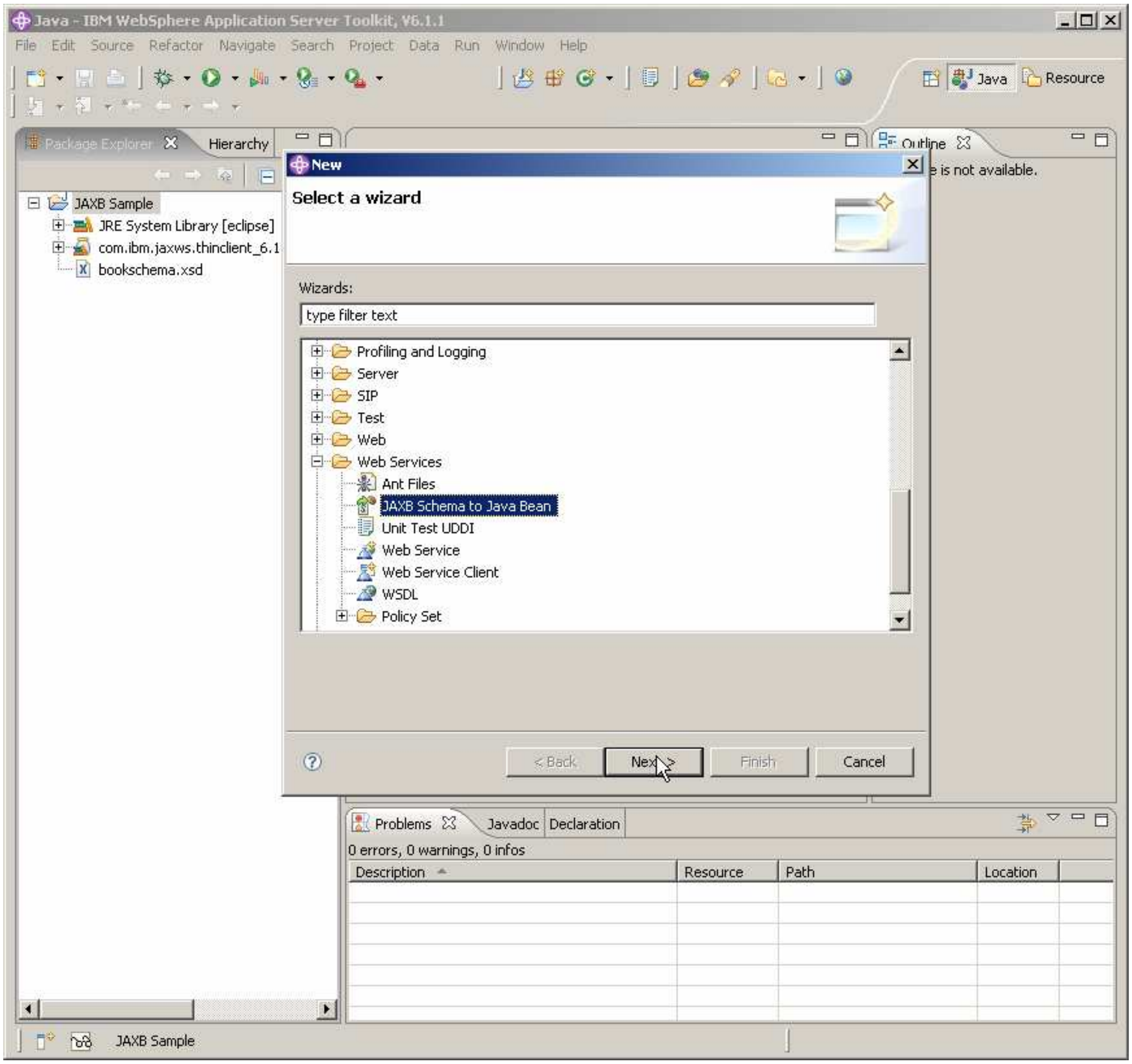
Part 4: Use the wizard to generate JAXB support files

In this section of the lab you will use the new JAXB wizards in the AST to create Java classes based on the schema you imported.

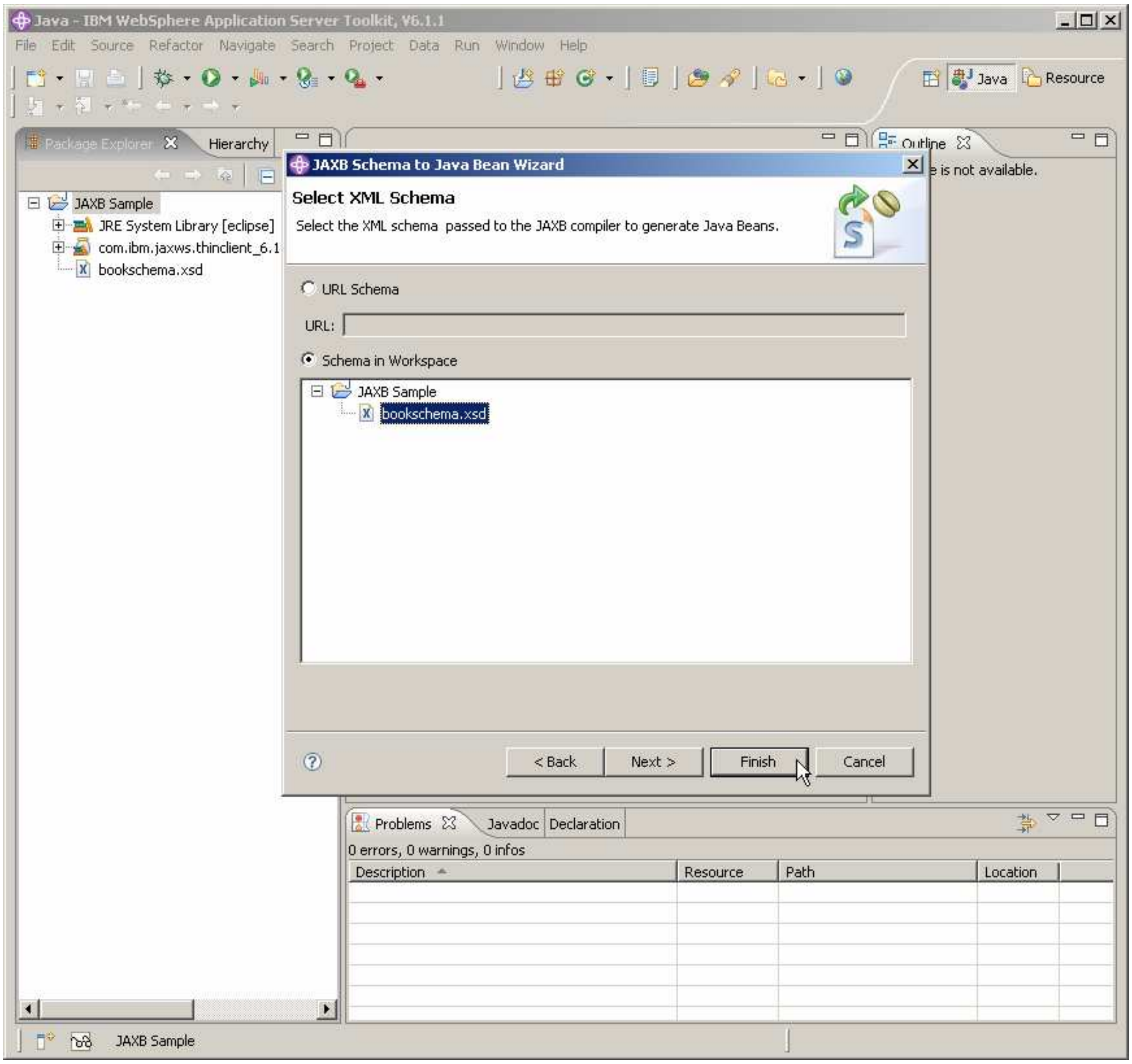
____ 4. Right click on the **JAXB Sample** project and select **New** then **Other**.



____ 5. Expand **Web Services** and choose **JAXB Schema to Java Bean** then click **Next**.



____ 6. In the **Select XML Schema** panel, select **Schema in Workspace.** Then expand the **JAXB Sample** project and select **bookschema.xsd.** Click **Finish**.



JAXB Schema to Java Bean Wizard

Select XML Schema

Select the XML schema passed to the JAXB compiler to generate Java Beans.

○ URL Schema

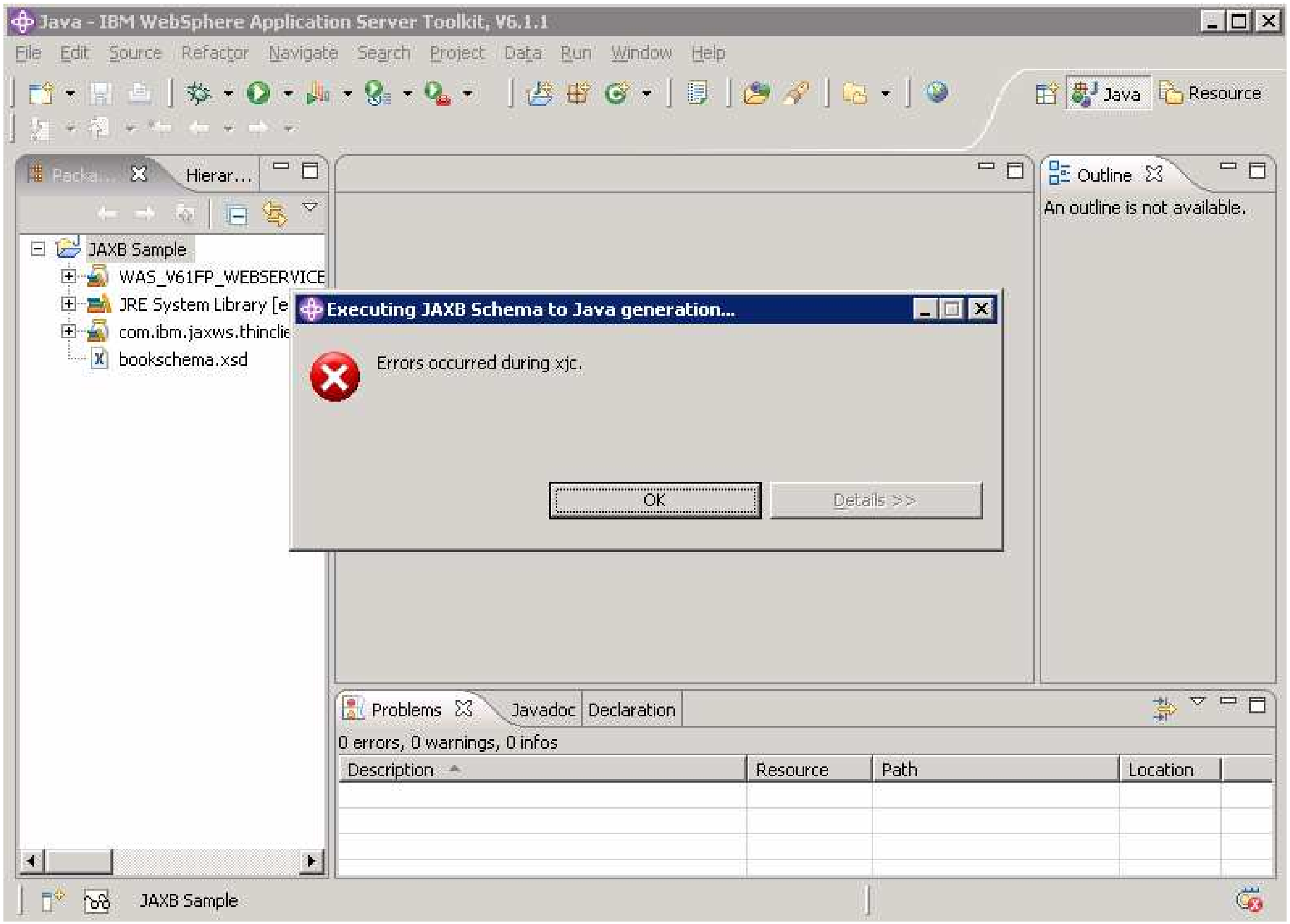
URL:

● Schema in Workspace

JAXB Sample
bookschema.xsd

Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Hierarchy
JAXB Sample
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1
bookschema.xsd
Outline
is not available.
< Back
Next >
Finish
Cancel
Problems
Javadoc
Declaration
0 errors, 0 warnings, 0 infos
Description
Resource
Path
Location
JAXB Sample

____ 7. There is a known defect at this time with the AST. An error will be generated. Click **OK.**



Executing JAXB Schema to Java generation...

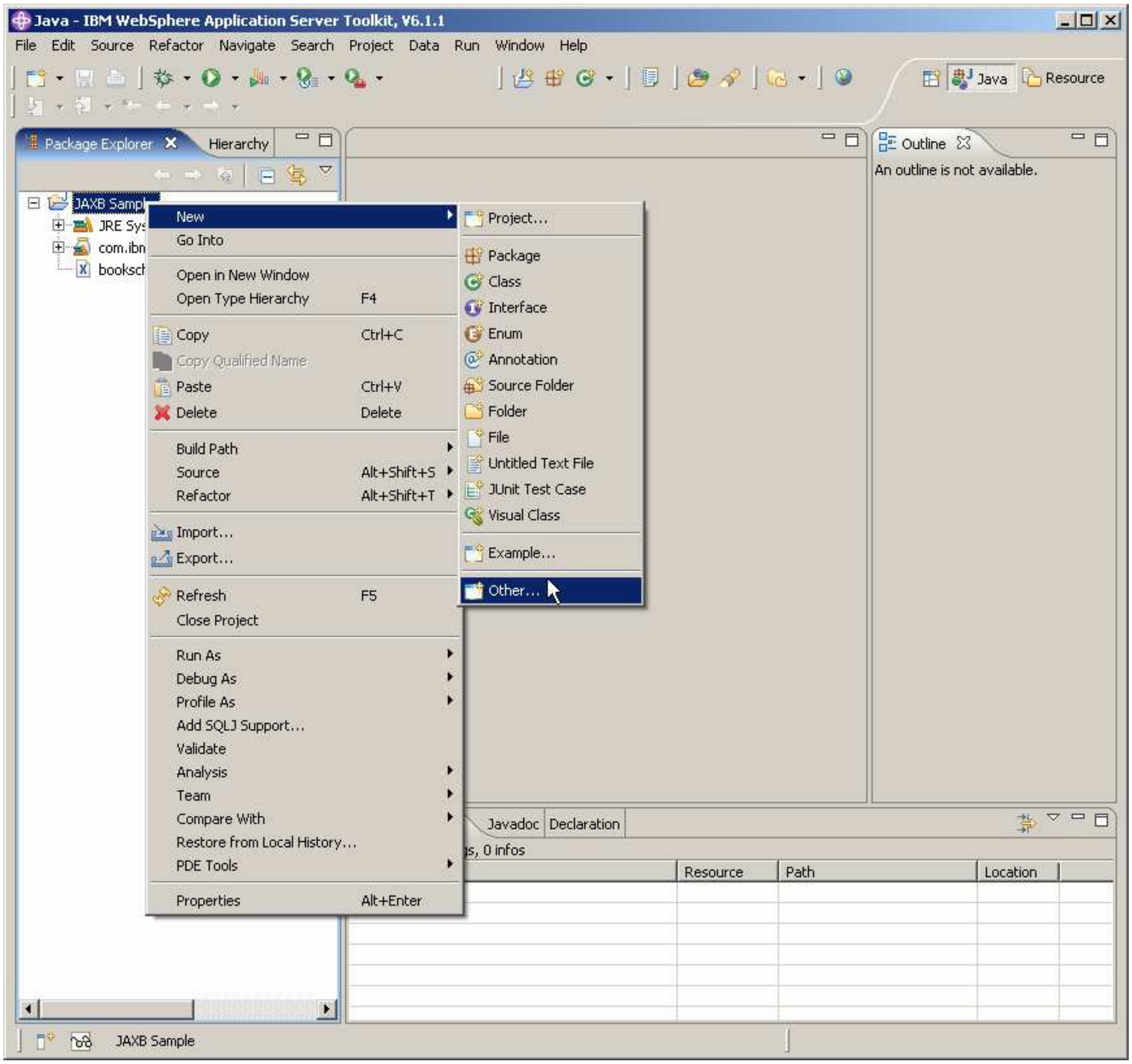
Errors occurred during xjc.

An outline is not available.

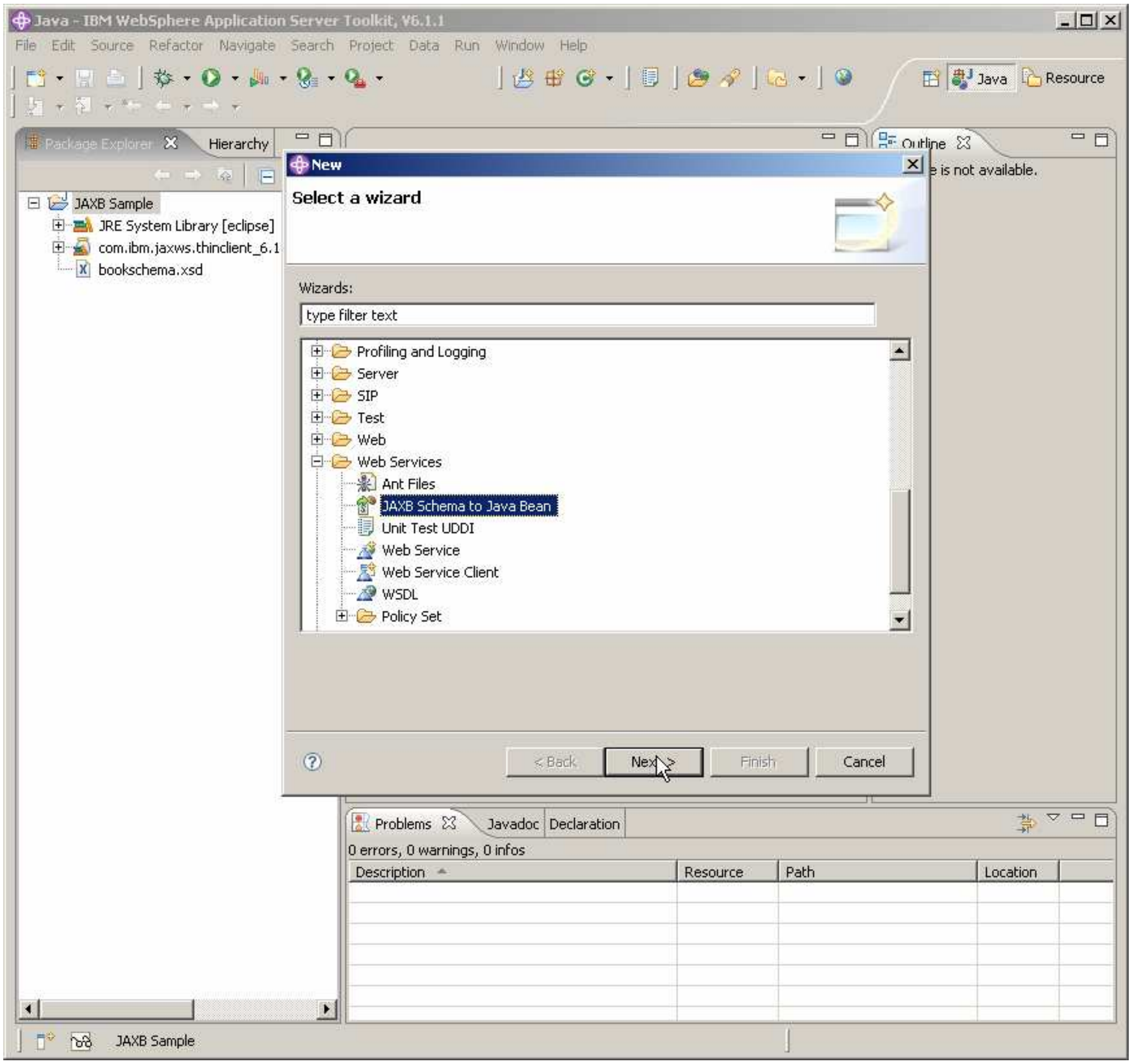
0 errors, 0 warnings, 0 infos

____ 8. If you don't encounter the error, move on to the next section.

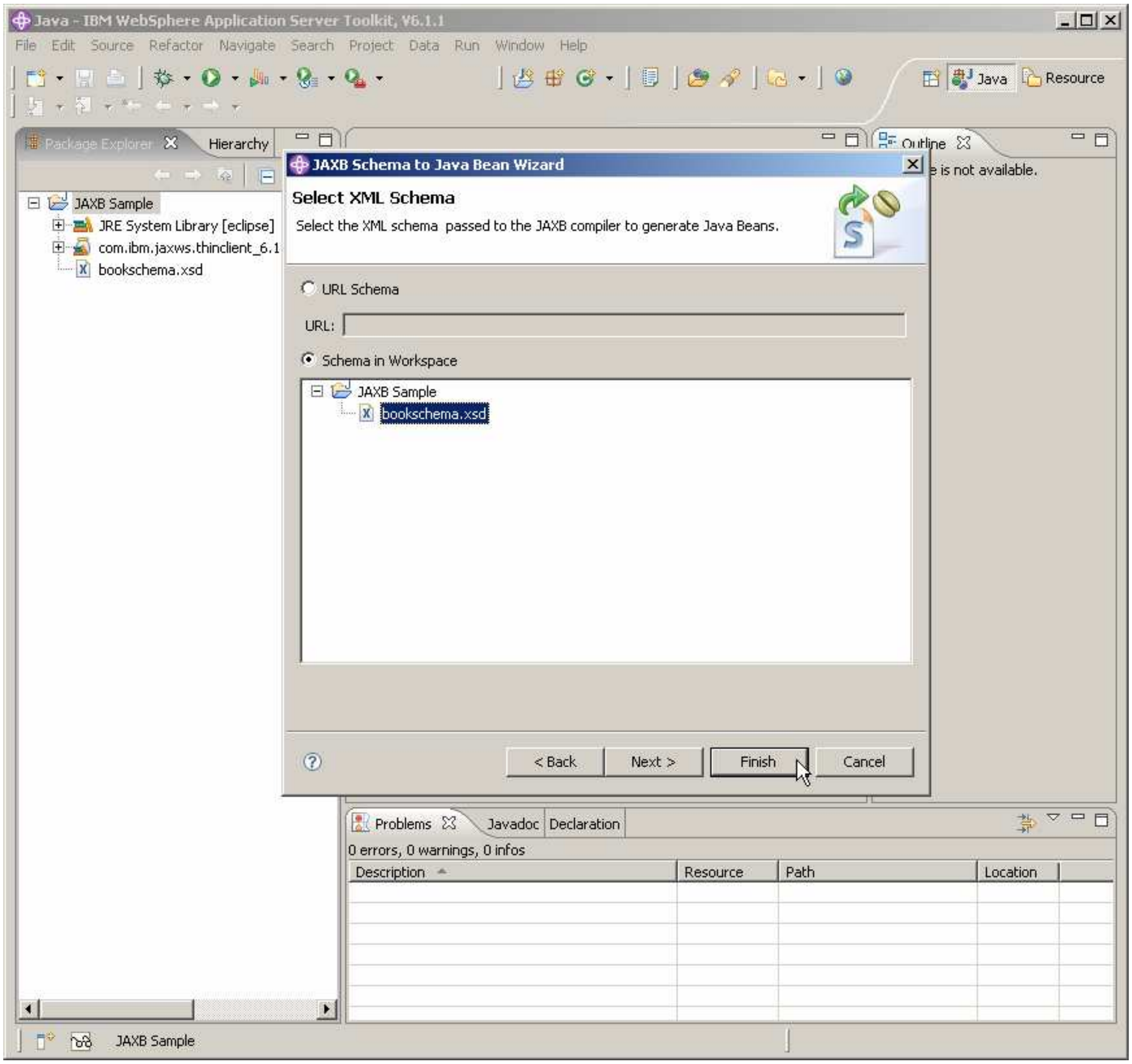
____ 9. Otherwise, run the wizard again. Right click on the **JAXB Sample** project and select **New** then **Other**.



____ 10. Expand **Web Services** and choose **JAXB Schema to Java Bean** then click **Next**.



____ 11. In the **Select XML Schema** panel, select **Schema in Workspace.** Then expand the **JAXB Sample** project and select **bookschema.xsd.** Click **Finish**.


Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Hierarchy
JAXB Sample
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1
bookschema.xsd
Outline
e is not available.
< Back
Next >
Finish
Cancel
Problems
Javadoc
Declaration
Description
Resource
Path
Location
JAXB Sample

JAXB Schema to Java Bean Wizard

Select XML Schema

Select the XML schema passed to the JAXB compiler to generate Java Beans.

○ URL Schema

URL:

● Schema in Workspace

- JAXB Sample
  - bookschema.xsd

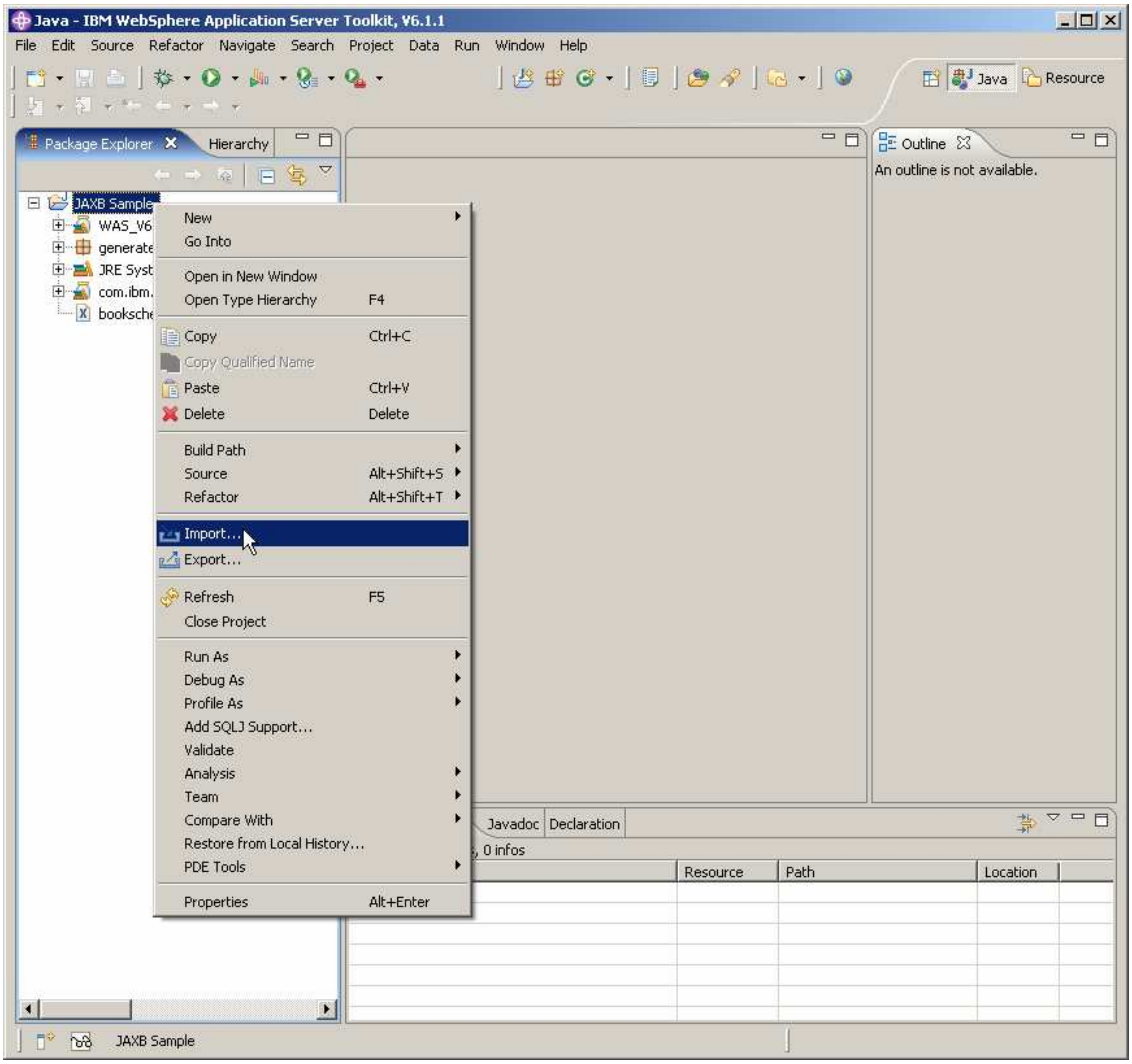
0 errors, 0 warnings, 0 infos

____ 12. The second time the wizard should run correctly, generating the appropriate classes.

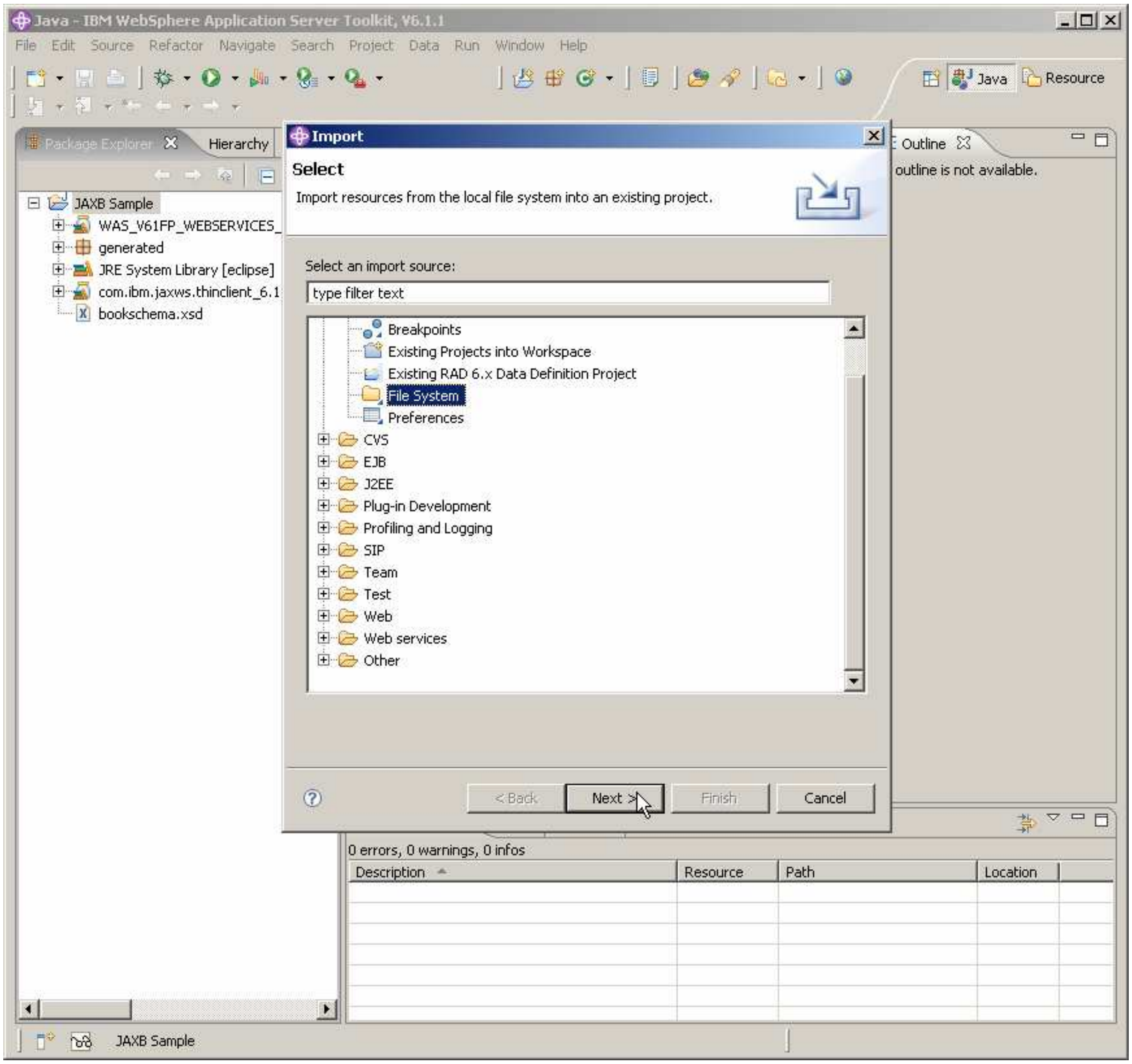
Part 5: Create the user code

Import code that leverages the generate schema.

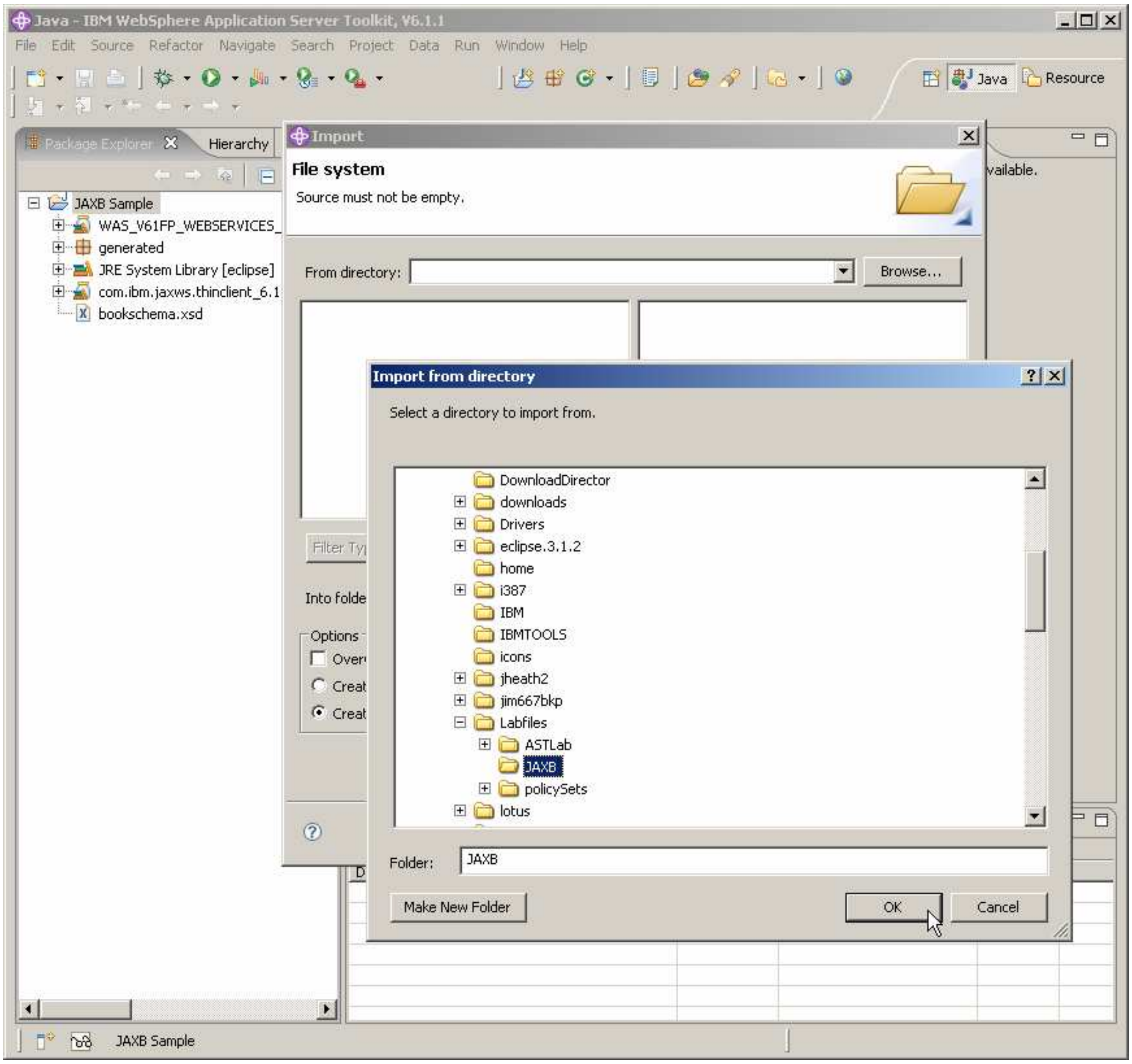
____ 13. Right click on the **JAXB Sample** project and select **Import**.



____ 14. Expand **General**, select **File System** and click **Next**.


Java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Hierarchy
JAXB Sample
WAS_V61FP_WEBSERVICES_
generated
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1
bookschema.xsd
Import
Select
Import resources from the local file system into an existing project.
Select an import source:
type filter text
Breakpoints
Existing Projects into Workspace
Existing RAD 6.x Data Definition Project
File System
Preferences
CVS
EJB
J2EE
Plug-in Development
Profiling and Logging
SIP
Team
Test
Web
Web services
Other
< Back
Next >
Finish
Cancel
Outline
outline is not available.
0 errors, 0 warnings, 0 infos
Description
Resource
Path
Location
JAXB Sample

____ 15. Browse to **C:\Labfiles\JAXB** and click **OK**.



Java - IBM WebSphere Application Server Toolkit, V6.1.1

Import

File system

Source must not be empty.

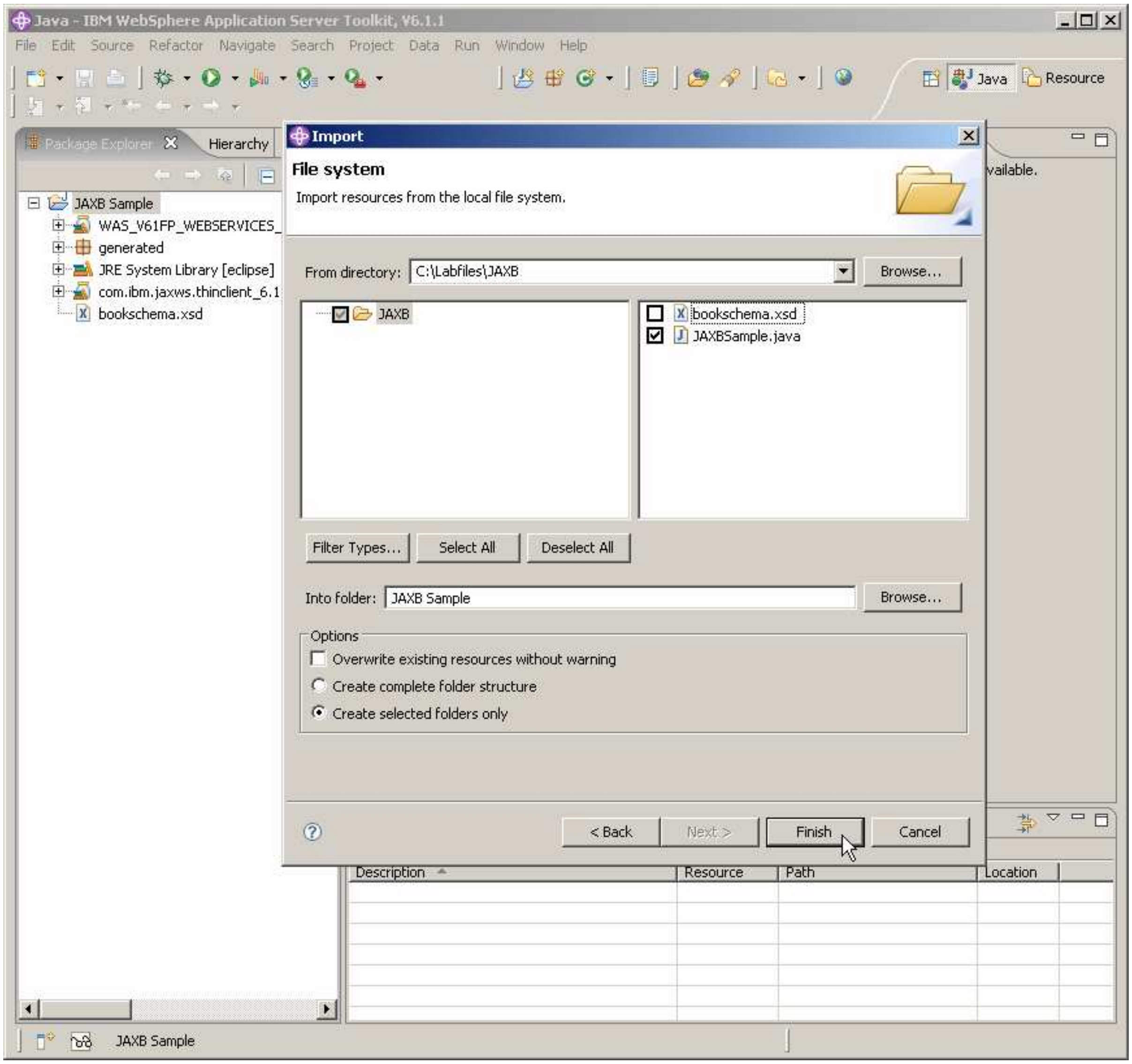
From directory:

Import from directory

Select a directory to import from.

Folder: JAXB

____ 16. Place a check mark beside **JAXBSample.java** and click **Finish**.



Import

File system

Import resources from the local file system.

From directory: C:\Labfiles\JAXB

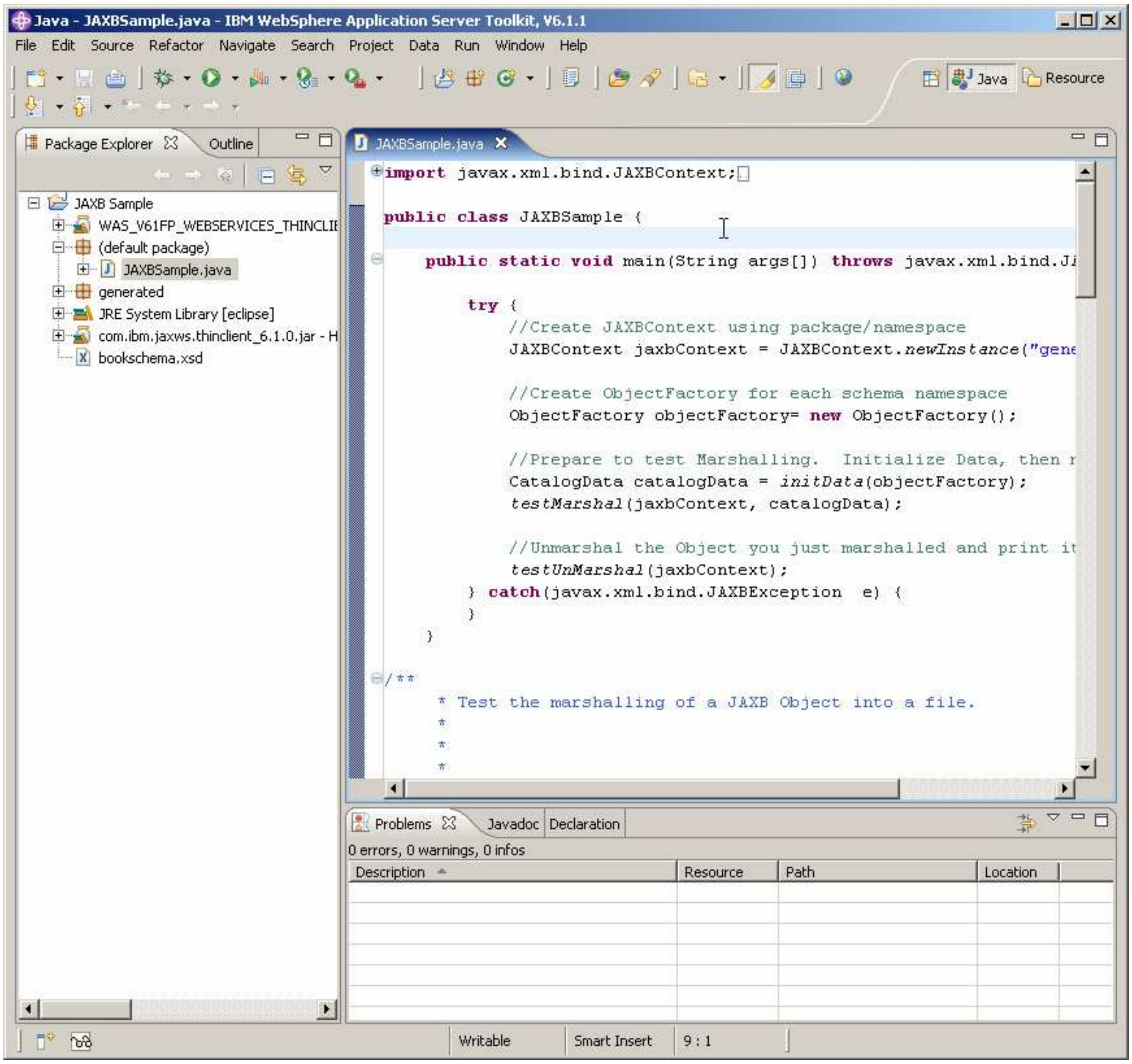
☑ JAXB

☐ bookschema.xsd
☑ JAXBSample.java

Into folder: JAXB Sample

Options
☐ Overwrite existing resources without warning
○ Create complete folder structure
● Create selected folders only

____ 17. The **(default package)** is created under **JAXB Sample**. Expand (default package) to find **JAXBSample.java**. Double-click **JAXBSample.java** to see the source.


Java - JAXBSample.java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Outline
JAXB Sample
WAS_V61FP_WEBSERVICES_THINCLIE
(default package)
JAXBSample.java
generated
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1.0.jar - H
bookschema.xsd
JAXBSample.java
Problems
Javadoc
Declaration
Writable
Smart Insert
9 : 1

```
import javax.xml.bind.JAXBContext;□

public class JAXBSample {

    public static void main(String args[]) throws javax.xml.bind.J

        try {
            //Create JAXBContext using package/namespace
            JAXBContext jaxbContext = JAXBContext.newInstance("gene

            //Create ObjectFactory for each schema namespace
            ObjectFactory objectFactory= new ObjectFactory();

            //Prepare to test Marshalling.  Initialize Data, then r
            CatalogData catalogData = initData(objectFactory);
            testMarshal(jaxbContext, catalogData);

            //Unmarshal the Object you just marshalled and print it
            testUnMarshal(jaxbContext);
        } catch(javax.xml.bind.JAXBException  e) {
        }
    }

/**
     * Test the marshalling of a JAXB Object into a file.
     *
     *
     *
```

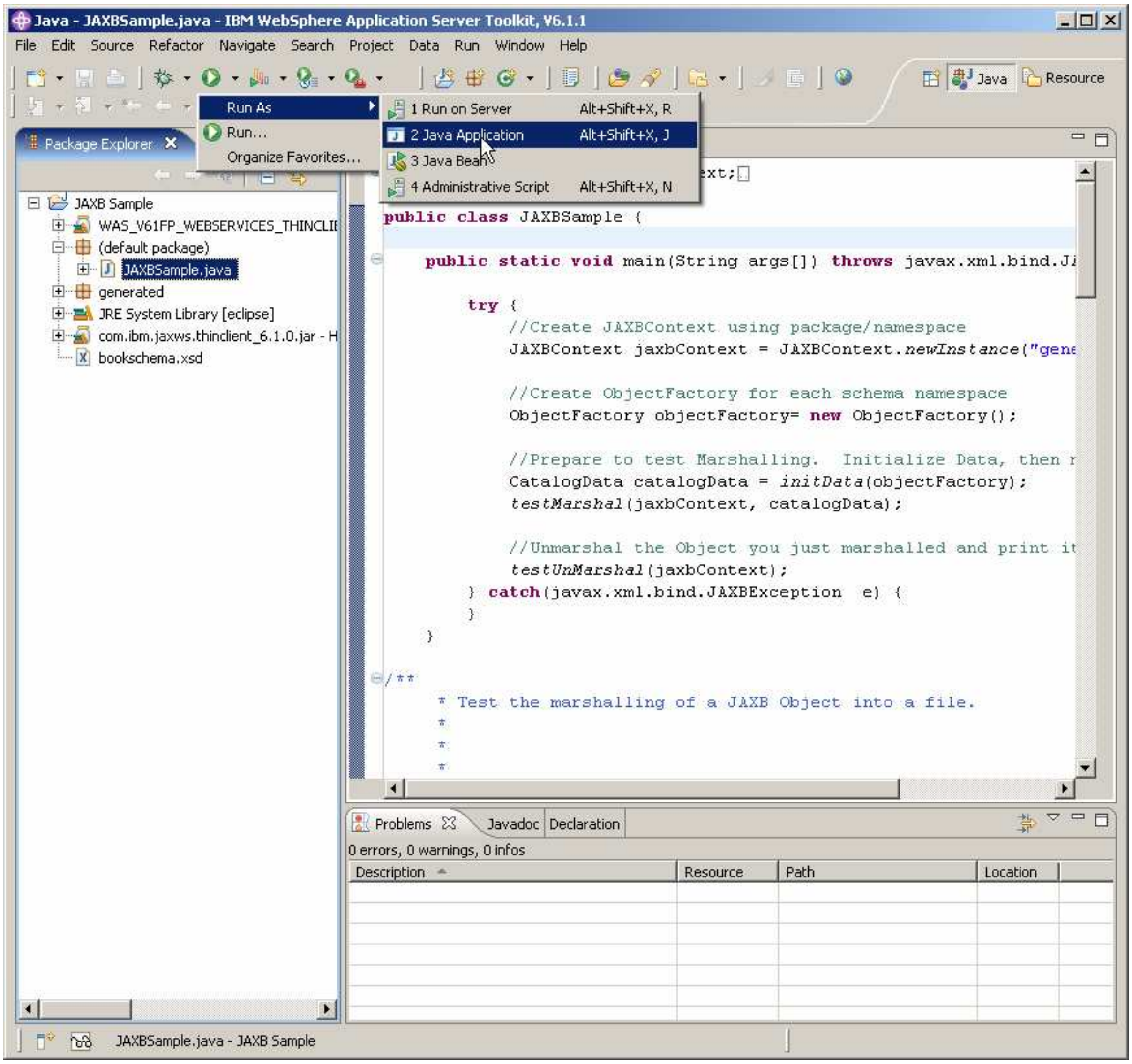
0 errors, 0 warnings, 0 infos

Description	Resource	Path	Location

Part 6: Run the user code

Now you will run the sample code provided that uses the classes generated from the schema.

____ 18. Run the JAXBSample program by clicking the **Run** icon, choosing **Run As**, and then choosing **Java Application**.

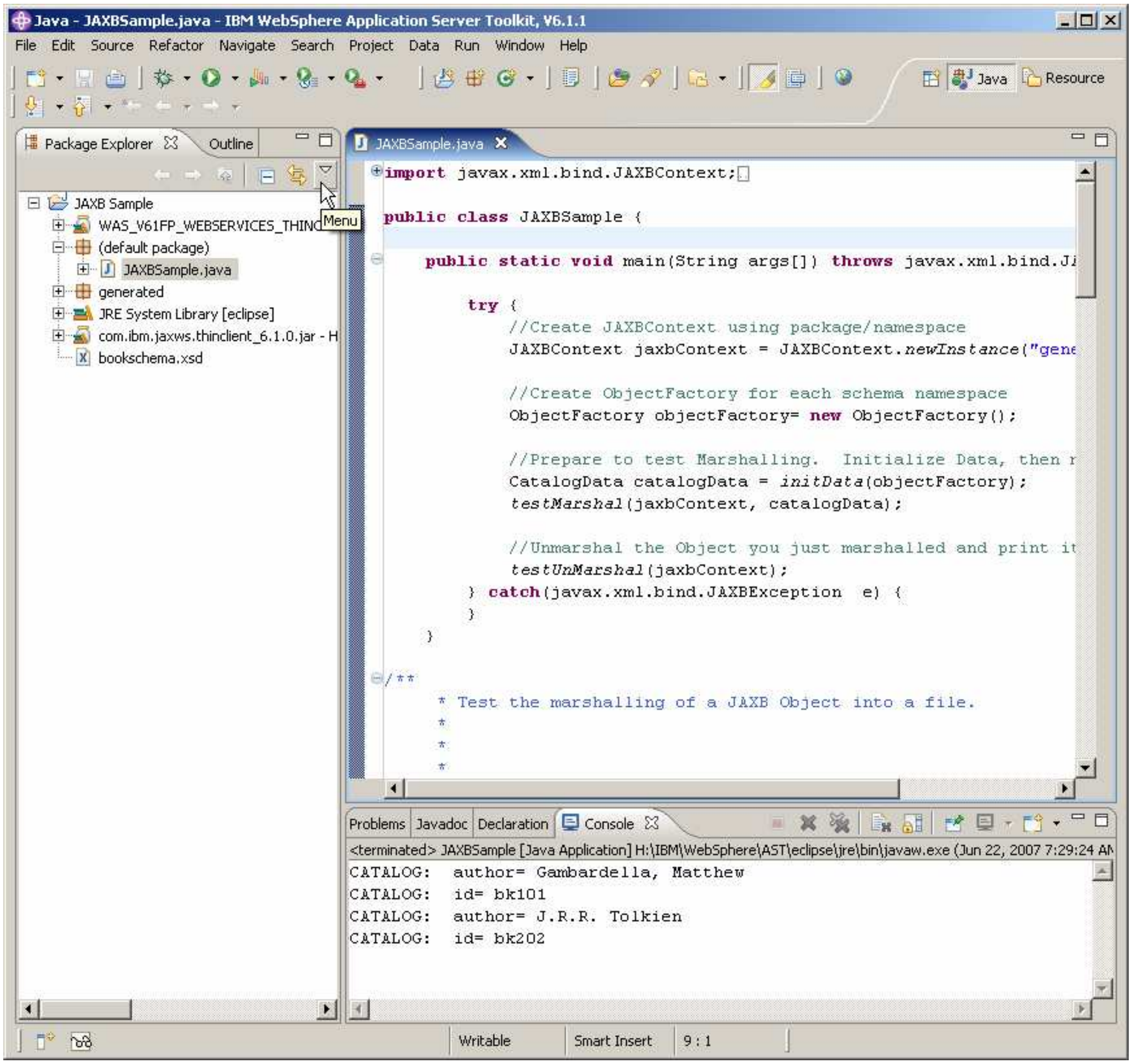

Java - JAXBSample.java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Run As
Run...
Organize Favorites...
1 Run on Server Alt+Shift+X, R
2 Java Application Alt+Shift+X, J
3 Java Bean
4 Administrative Script Alt+Shift+X, N
Package Explorer
JAXB Sample
WAS_V61FP_WEBSERVICES_THINCLIE
(default package)
JAXBSample.java
generated
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1.0.jar - H
bookschema.xsd
Problems
Javadoc
Declaration
JAXBSample.java - JAXB Sample

```
ext;
public class JAXBSample {

    public static void main(String args[]) throws javax.xml.bind.J

        try {
            //Create JAXBContext using package/namespace
            JAXBContext jaxbContext = JAXBContext.newInstance("gene

            //Create ObjectFactory for each schema namespace
            ObjectFactory objectFactory= new ObjectFactory();

            //Prepare to test Marshalling.  Initialize Data, then r
            CatalogData catalogData = initData(objectFactory);
            testMarshal(jaxbContext, catalogData);

            //Unmarshal the Object you just marshalled and print it
            testUnMarshal(jaxbContext);
        } catch(javax.xml.bind.JAXBException  e) {
        }
    }

/**
     * Test the marshalling of a JAXB Object into a file.
     *
     *
     *
```

0 errors, 0 warnings, 0 infos

Description	Resource	Path	Location

____ 19. In the Console panel of the AST, look for the output shown below.


Java - JAXBSample.java - IBM WebSphere Application Server Toolkit, V6.1.1
File Edit Source Refactor Navigate Search Project Data Run Window Help
Java
Resource
Package Explorer
Outline
Menu
JAXB Sample
WAS_V61FP_WEBSERVICES_THIN
(default package)
JAXBSample.java
generated
JRE System Library [eclipse]
com.ibm.jaxws.thinclient_6.1.0.jar - H
bookschema.xsd
JAXBSample.java
Problems
Javadoc
Declaration
Console
Writable
Smart Insert
9 : 1

```
import javax.xml.bind.JAXBContext;

public class JAXBSample {

    public static void main(String args[]) throws javax.xml.bind.J

        try {
            //Create JAXBContext using package/namespace
            JAXBContext jaxbContext = JAXBContext.newInstance("gene

            //Create ObjectFactory for each schema namespace
            ObjectFactory objectFactory= new ObjectFactory();

            //Prepare to test Marshalling.  Initialize Data, then r
            CatalogData catalogData = initData(objectFactory);
            testMarshal(jaxbContext, catalogData);

            //Unmarshal the Object you just marshalled and print it
            testUnMarshal(jaxbContext);
        } catch(javax.xml.bind.JAXBException  e) {
        }
    }

/**
  * Test the marshalling of a JAXB Object into a file.
  *
  *
  *
```

```
<terminated> JAXBSample [Java Application] H:\IBM\WebSphere\AST\eclipse\jre\bin\javaw.exe (Jun 22, 2007 7:29:24 AM
CATALOG:  author= Gambardella, Matthew
CATALOG:  id= bk101
CATALOG:  author= J.R.R. Tolkien
CATALOG:  id= bk202
```

Summary

In this lab, you created a marshaller and unmarshaller using the bindings created by JAXB tools in the AST. You performed the following steps to do this:

1. Created a Java Project in the AST and set a property to include a needed jar file
2. Imported an XML Schema file into the Java Project
3. Used the AST wizard to generate JAXB support files
4. Created user code to invoke the JAXB supplied methods
5. Ran the user code to confirm the correctness

This page is left intentionally blank.