

Developing Java in Eclipse: Part I of III SHARE Orlando, February 2008

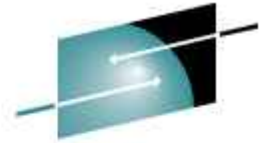
Oliver Fenton

Java Technology Center, IBM Hursley Labs,
Winchester, UK

Session: 8358



Trademarks

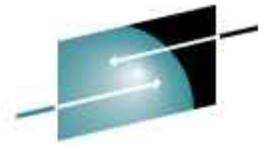


SHARE
Technology • Connections • Results

IBM, AIX, CICS, DB2, IMS, z/OS, OS/390, S/390, System/390, VisualAge, WebSphere Application Server, WebSphere Studio, z/VM - are trademarks or registered trademarks of the IBM Corporation

Sun, Sun Microsystems, JavaSoft, Java, JavaBeans, JDK, Java 2 Micro Edition, J2ME, Java 2 Standard Edition, J2SE, Java 2 Enterprise Edition, J2EE - are trademarks or registered trademarks of Sun Microsystems Inc.

Agenda



SHARE
Technology • Connections • Results

❖ Introduction

- Eclipse Overview and History
- Eclipse Architecture and plug-ins
- Perspectives
- Projects

❖ The Java Perspective

❖ Eclipse Java Projects

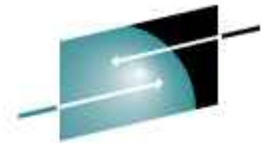
❖ Exercise

❖ Further Help and Additional Resources

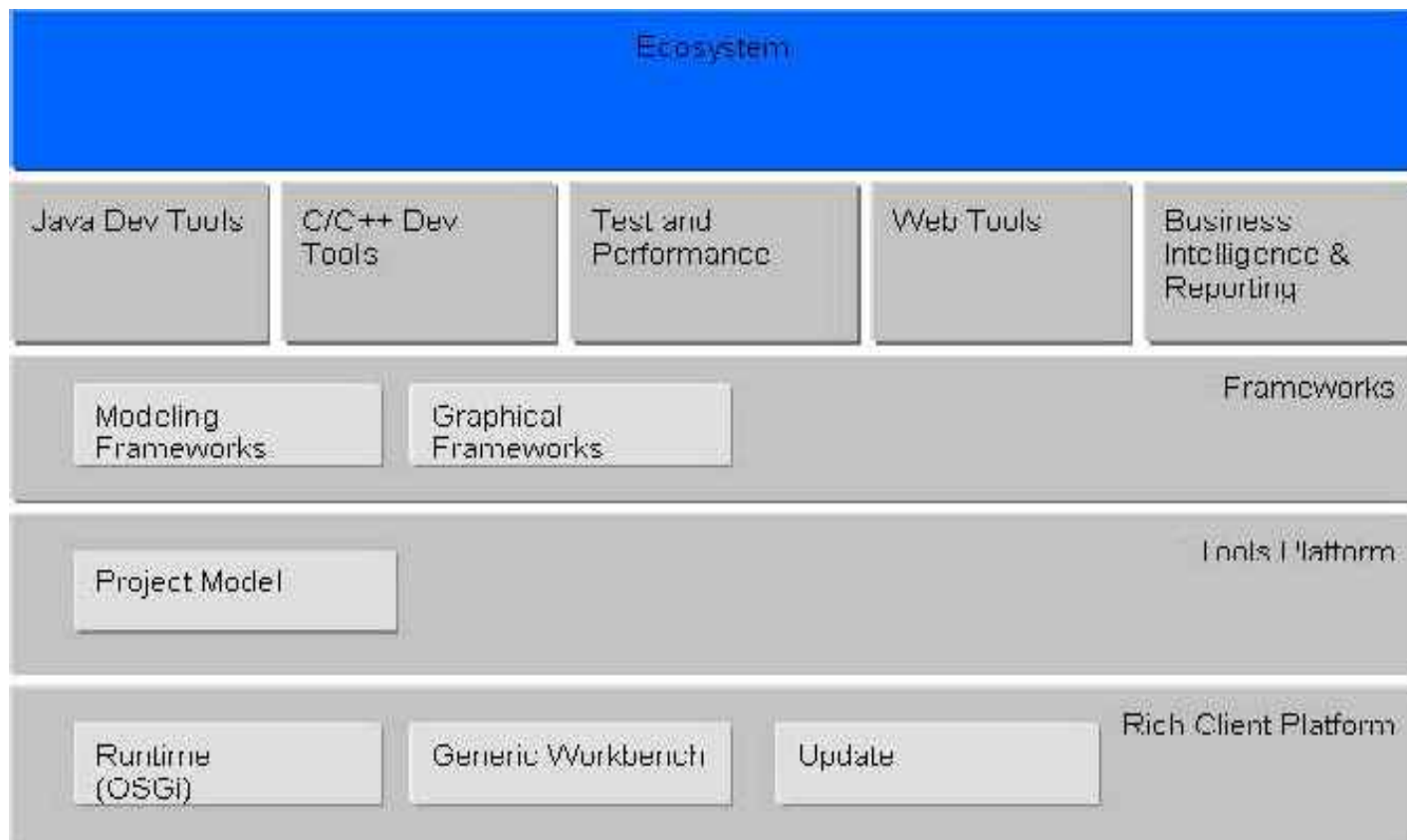
Eclipse Overview and History

- ❖ Eclipse is an extensible Integrated Development Environment (IDE) based on a plug-in architecture
- ❖ Eclipse is open source, released under a common public license (CPL)
- ❖ Originally developed by IBM, version 1.0 being released in November 2001, Eclipse is now under the stewardship of an independent consortium including: IBM, SAP, Sybase, Fujitsu, HP, BEA...
- ❖ Available as a free download from www.eclipse.org

Uses of Eclipse



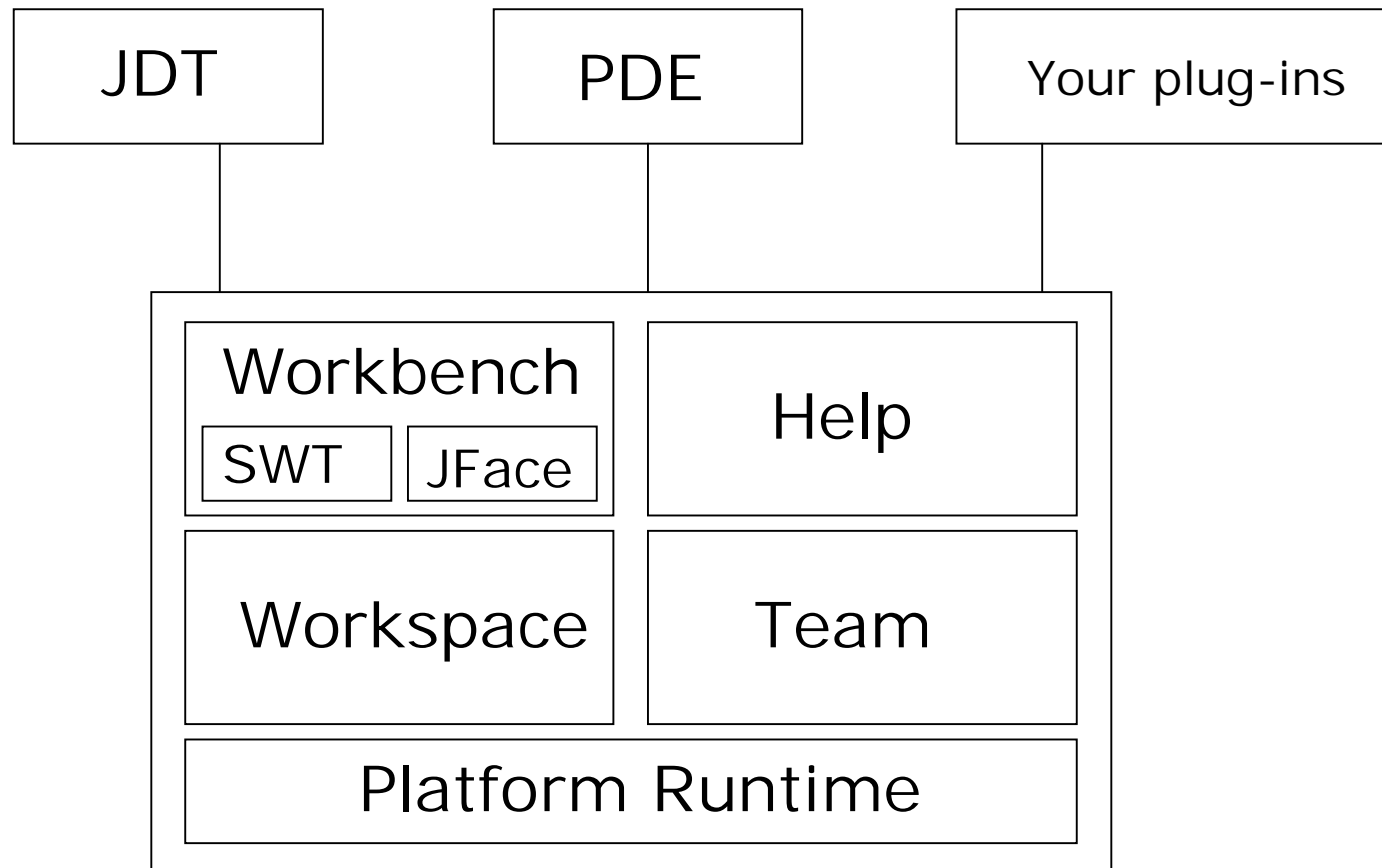
SHARE
Technology • Connections • Results





SHARE
Technology • Connections • Results

Eclipse Architecture



Plug In's

- ❖ Eclipse is a framework which enables plug-in operability and interoperability
- ❖ Click on 'Help>About Eclipse SDK' then 'Plug-in Details' to see currently active Plug-ins



- ❖ Eclipse plug-ins are available from a number of sources including www.eclipseplugincentral.com

Perspectives

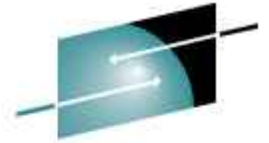


- ❖ Eclipse perspective is a pre-selected set of views arranged in a predetermined way.
- ❖ Included with Eclipse are several pre-customized perspectives including:
 - Java
 - Debug – session 8355 (Remote z/OS Debugging)
 - Testing – session 8384 (JUnit & Test Driven Development)
 - Plug-in Development
 - Team Synchronizing

Projects



- ❖ Eclipse projects contain folders and files like in a hierarchical file system
- ❖ An Eclipse project is a collection of resources (including code) that combine to make an application



SHARE
Technology • Connections • Results

Agenda

❖ Introduction

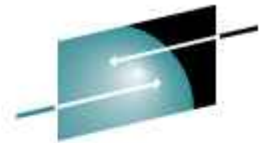
- Eclipse Overview and History
- Eclipse Architecture and plug-ins
- Perspectives
- Projects

❖ The Java Perspective

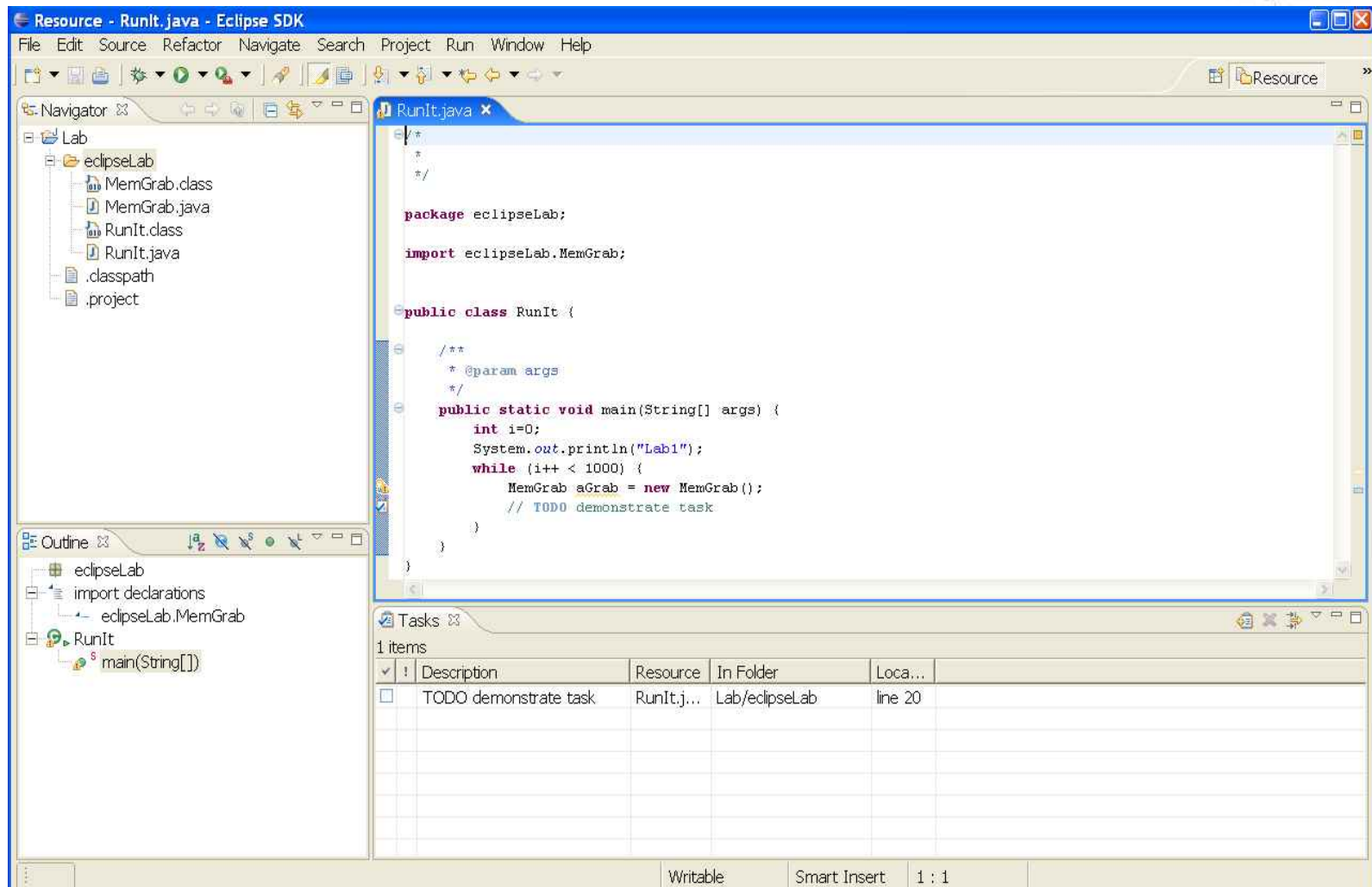
❖ Eclipse Java Projects

❖ Exercise

❖ Further Help and Additional Resources



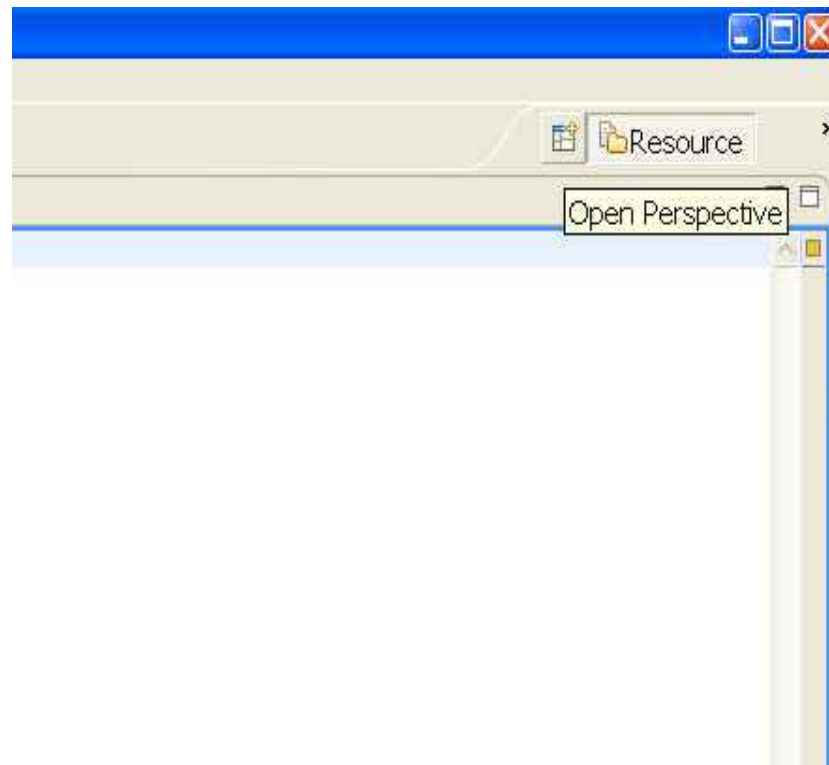
Perspectives and Eclipse 1



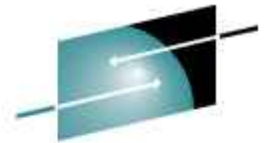
Perspectives and Eclipse 2



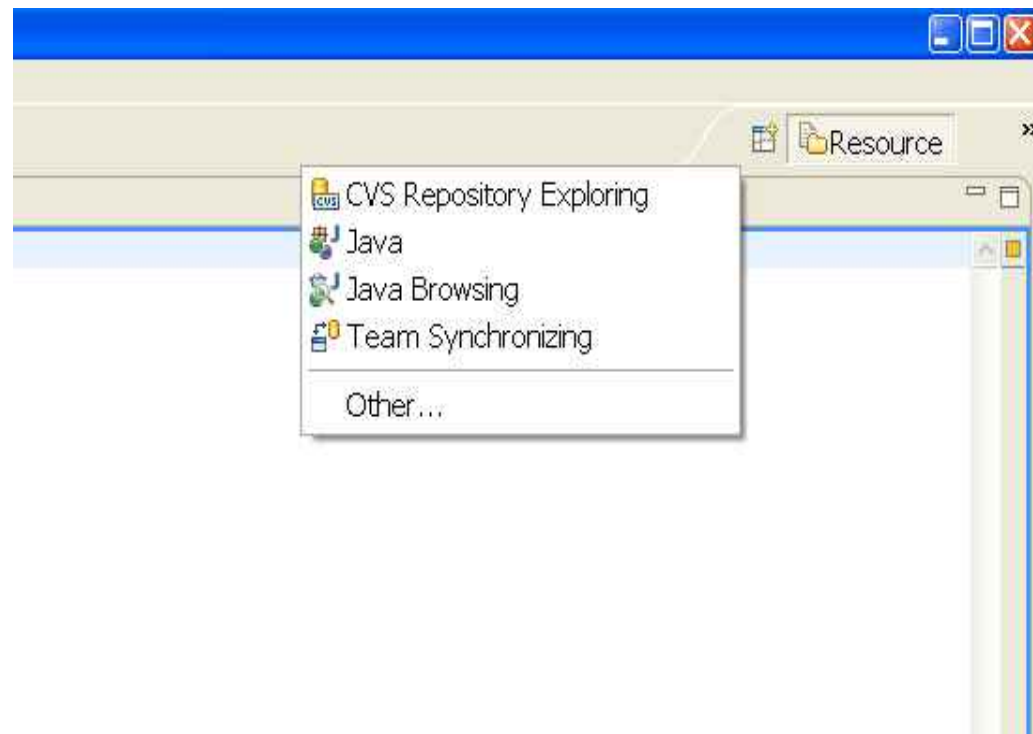
SHARE
Technology • Connections • Results



Perspectives and Eclipse 3



SHARE
Technology • Connections • Results





SHARE
Technology • Connections • Results

Java Perspective

Java - RunIt.java - Eclipse SDK

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- Lab
 - eclipseLab
 - MemGrab.java
 - MemGrab
 - MemGrab()
 - RunIt.java
 - RunIt
 - main(String[])
 - JRE System Library [Java14]

RunIt.java

```
/*
 *
 */
package eclipseLab;

import eclipseLab.MemGrab;

public class RunIt {

    /**
     * @param args
     */
    public static void main(String[] args) {
        int i=0;
        System.out.println("Lab1");
        while (i++ < 1000) {
            MemGrab aGrab = new MemGrab();
            // TODO demonstrate task
        }
    }
}
```

Outline

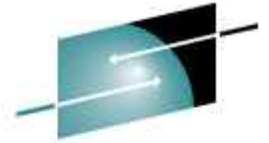
- eclipseLab
 - import declarations
 - eclipseLab.MemGrab
 - RunIt
 - main(String[])

Problems

0 errors, 2 warnings, 0 infos

Description	Resource	In Folder	Loca...
The local variable buf is ne...	MemGr...	Lab/eclipseLab	line 15
The local variable aGrab is...	RunIt.j...	Lab/eclipseLab	line 19

eclipseLab.RunIt.java - Lab



SHARE
Technology • Connections • Results

Agenda

❖ Introduction

- Eclipse Overview and History
- Eclipse Architecture and plug-ins
- Perspectives
- Projects

❖ The Java Perspective

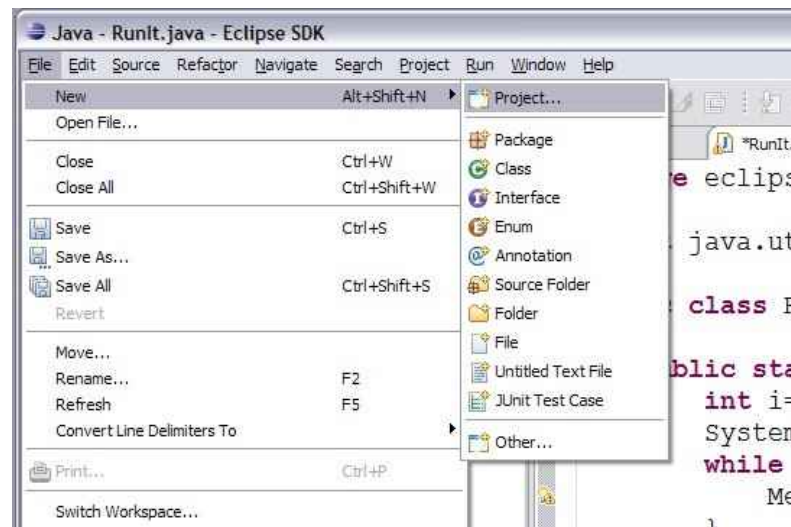
❖ Eclipse Java Projects

❖ Exercise

❖ Further Help and Additional Resources

Java Projects

- ❖ An Eclipse project is a collection of code and resources that combine to make an application
- ❖ Each project within a workspace can have different settings




Creating a New Java Project 1



SHARE
Technology • Connections • Results



Creating a New Java Project 2



The image shows the 'New Java Project' dialog box in an IDE. It has a title bar with a question mark icon and a close button. The main title is 'Create a Java project' with a subtitle 'Create a Java project in the workspace or in an external location.' and a folder icon. The 'Project name' field contains 'NewProject'. The 'Contents' section has two radio buttons: 'Create new project in workspace' (selected) and 'Create project from existing source'. The 'Directory' field shows 'C:\tools\workspaces\ eclipse_3.2\NewProject' with a 'Browse...' button. The 'JRE' section has two radio buttons: 'Use default JRE (Currently 'j9vmwi3223-20070426-sdk')' (selected) and 'Use a project specific JRE:' with a dropdown menu showing 'jre1.6.0_02' and a 'Configure JREs...' link. The 'Project layout' section has two radio buttons: 'Use project folder as root for sources and class files' (selected) and 'Create separate source and output folders' with a 'Configure default...' link. At the bottom are buttons for '< Back', 'Next >', 'Finish', and 'Cancel', along with a help icon.

New Java Project

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

Project name:

Contents

☒ Create new project in workspace
☐ Create project from existing source

Directory: [Browse...](#)

JRE

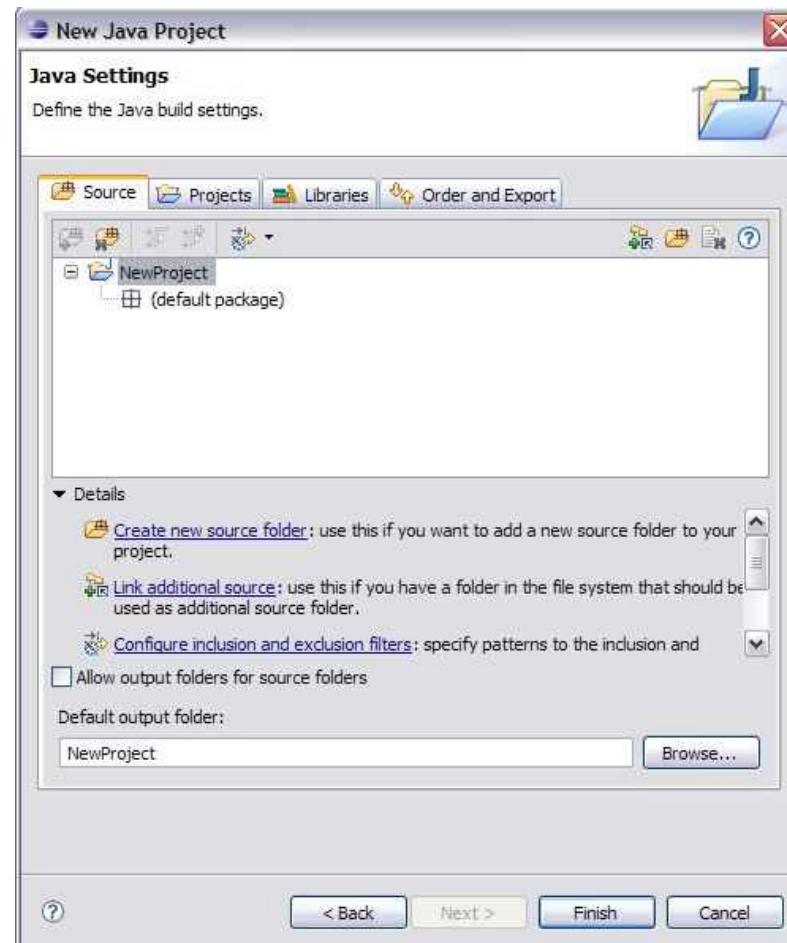
☒ Use default JRE (Currently 'j9vmwi3223-20070426-sdk') [Configure JREs...](#)
☐ Use a project specific JRE:

Project layout

☒ Use project folder as root for sources and class files
☐ Create separate source and output folders [Configure default...](#)

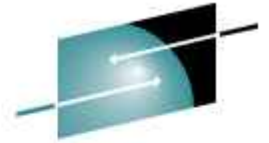
[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

Creating a New Java Project 3



Exercise

- ❖ Creating a project, package and class
 - Example program is CountPrinter.java
- ❖ Code assist, identifying errors, code completion
- ❖ Running code



SHARE
Technology • Connections • Results

Some Cool Features of Eclipse

- ❖ Automatic building
- ❖ Classpath resolution
- ❖ Real-time error checking
- ❖ Code assist
- ❖ Quick-Fix
- ❖ Refactoring
- ❖ Declarations & References

Review



SHARE
Technology • Connections • Results

❖ Introduction

- Eclipse Overview and History
- Eclipse Architecture and plug-ins
- Perspectives
- Projects

❖ The Java Perspective

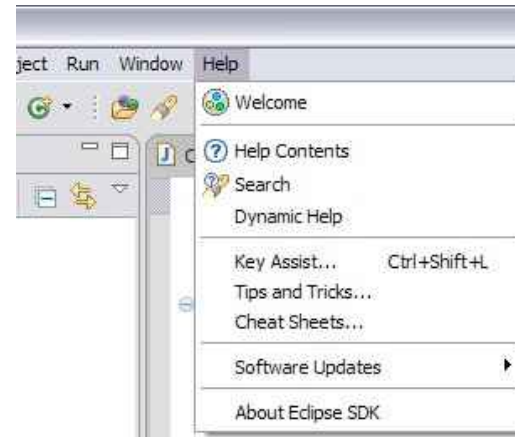
❖ Eclipse Java Projects

❖ Exercise

❖ Further Help and Additional Resources

Further Help and Additional Resources

❖ Eclipse in-built help



- ❖ www.eclipse.org
- ❖ www.eclipseplugincentral.com
- ❖ www.google.com