Secure Programming – Assignment 1

Rathna Sindura, Chikkam | rxc3518 | 1001553518

Part 1

Manual Analysis

- On Line 22, PORT variable is initialized with a static value. Instead of using static values, the variable should be kept separately in a constants.java file and can be used in the SimpleWebServer class. Using hard-coded values within a class is not an acceptable practice.
- On line 29, Static object dServerSocket is instantiated separately, which is not an acceptable practice either, because static objects are supposed to be initialized only once.
- Exception handling is too generic in the run() method. Try, catch blocks are not present.
- End statement is not invoked for the while loop in run() method.
- In line 62, while parsing the HTTP request, StringTokenizer class constructor takes two string parameters, of which one is the http request and other is a space delimiter. Instead of sending a space directly through the constructor, a better approach would be to initialize the delimiter and then use the delimiter variable.
- Like run() method, try catch blocks are not added in this processRequest method, implementing generic exception handling is not an acceptable practice.
- serveFile method contains try catch blocks, but the catch block do not log the actual exception message "e.message" but only logs the generic exception. Apart from file not being present, exception can be that appropriate permissions are not provided on the specified location to read the file. Hence it is important to log the exception.
- FileReader object fr is not closed in this serveFile method, which is not an acceptable practice as it leads to memory leak.
- Main method too does not contain try catch blocks and exception handling is poorly implemented.

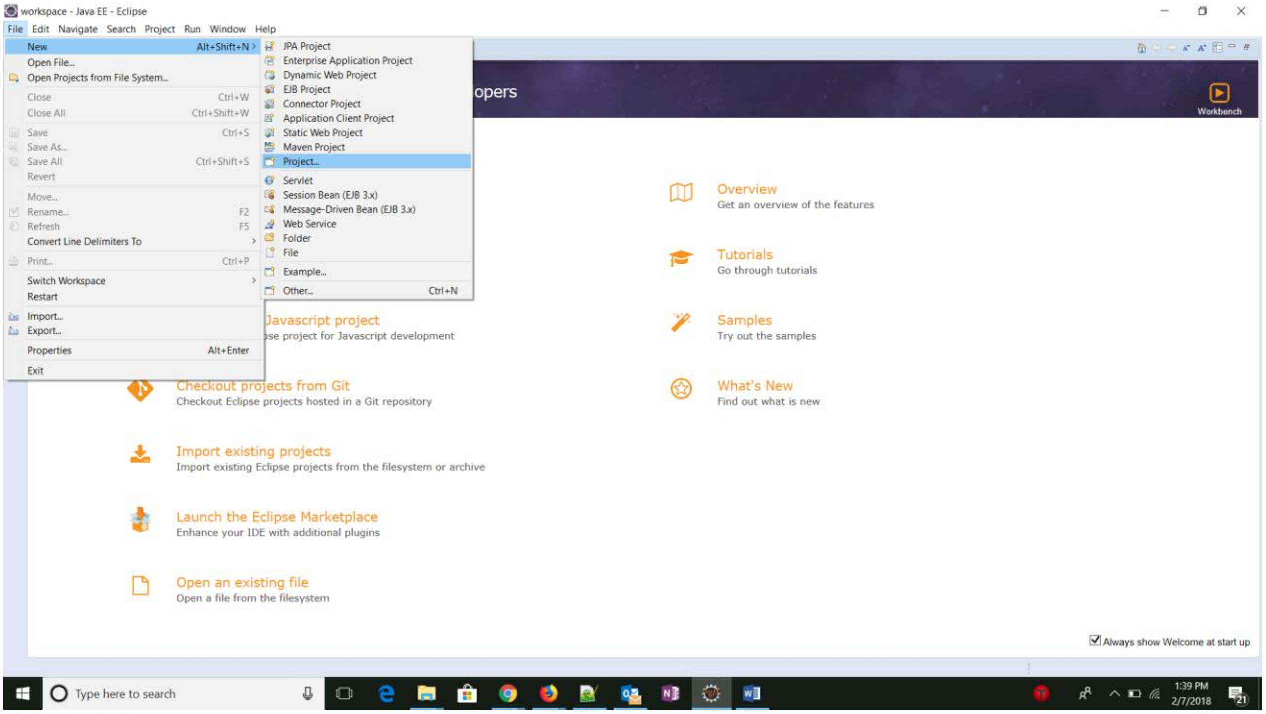
Tool Choices/ Versions

The tools I have used to test the code are:

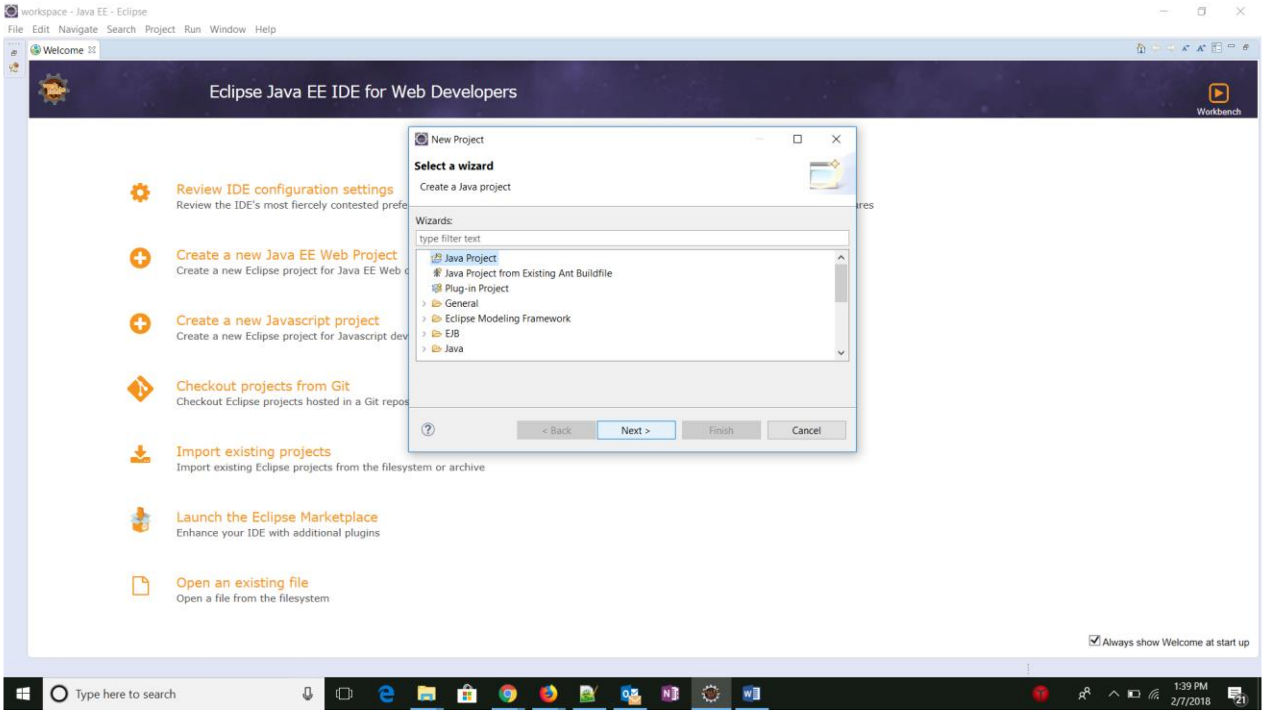
- Eclipse Neon 3, Java JDK 8
- Findbugs – 3.0.1
- SonarLint – 3.3.1

Tool Invocation Process

1. Using Eclipse, build a project and package named SimpleWebServer and add the SimpleWebServer class as shown in the steps from (a) to (h) below:
2.

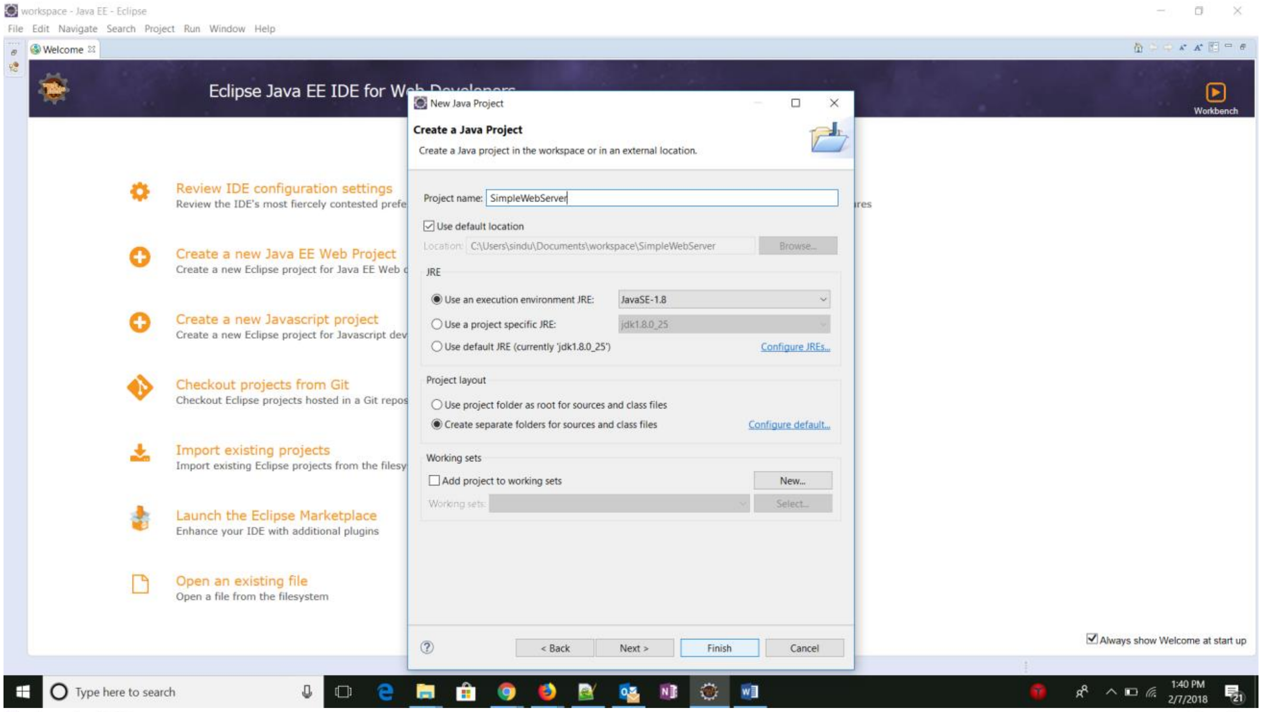

workspace - Java EE - Eclipse
File Edit Navigate Search Project Run Window Help
New Alt+Shift+N >
Open File...
Open Projects from File System...
Close Ctrl+W
Close All Ctrl+Shift+W
Save Ctrl+S
Save As...
Save All Ctrl+Shift+S
Revert
Move...
Rename... F2
Refresh F5
Convert Line Delimiters To >
Print... Ctrl+P
Switch Workspace >
Restart
Import...
Export...
Properties Alt+Enter
Exit
JPA Project
Enterprise Application Project
Dynamic Web Project
EJB Project
Connector Project
Application Client Project
Static Web Project
Maven Project
Project...
Servlet
Session Bean (EJB 3.x)
Message-Driven Bean (EJB 3.x)
Web Service
Folder
File
Example...
Other... Ctrl+N
opers
Workbench
Overview
Get an overview of the features
Tutorials
Go through tutorials
Samples
Try out the samples
What's New
Find out what is new
Javascript project
se project for Javascript development
Checkout projects from Git
Checkout Eclipse projects hosted in a Git repository
Import existing projects
Import existing Eclipse projects from the filesystem or archive
Launch the Eclipse Marketplace
Enhance your IDE with additional plugins
Open an existing file
Open a file from the filesystem
☑ Always show Welcome at start up
Type here to search
1:39 PM
2/7/2018

a.


workspace - Java EE - Eclipse
File Edit Navigate Search Project Run Window Help
Welcome
Eclipse Java EE IDE for Web Developers
Workbench
Review IDE configuration settings
Review the IDE's most fiercely contested prefe
Create a new Java EE Web Project
Create a new Eclipse project for Java EE Web
Create a new Javascript project
Create a new Eclipse project for Javascript dev
Checkout projects from Git
Checkout Eclipse projects hosted in a Git repos
Import existing projects
Import existing Eclipse projects from the filesystem or archive
Launch the Eclipse Marketplace
Enhance your IDE with additional plugins
Open an existing file
Open a file from the filesystem
ures
New Project
Select a wizard
Create a Java project
Wizards:
type filter text
Java Project
Java Project from Existing Ant Buildfile
Plug-in Project
General
Eclipse Modeling Framework
EJB
Java
< Back
Next >
Finish
Cancel
☑ Always show Welcome at start up
Type here to search
1:39 PM
2/7/2018

b.

c.



New Java Project

Create a Java Project

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

Project name: SimpleWebServer

☑ Use default location

Location: C:\Users\sindu\Documents\workspace\SimpleWebServer

JRE

● Use an execution environment JRE: JavaSE-1.8

○ Use a project specific JRE: jdk1.8.0_25

○ Use default JRE (currently 'jdk1.8.0_25')

Project layout

○ Use project folder as root for sources and class files

● Create separate folders for sources and class files

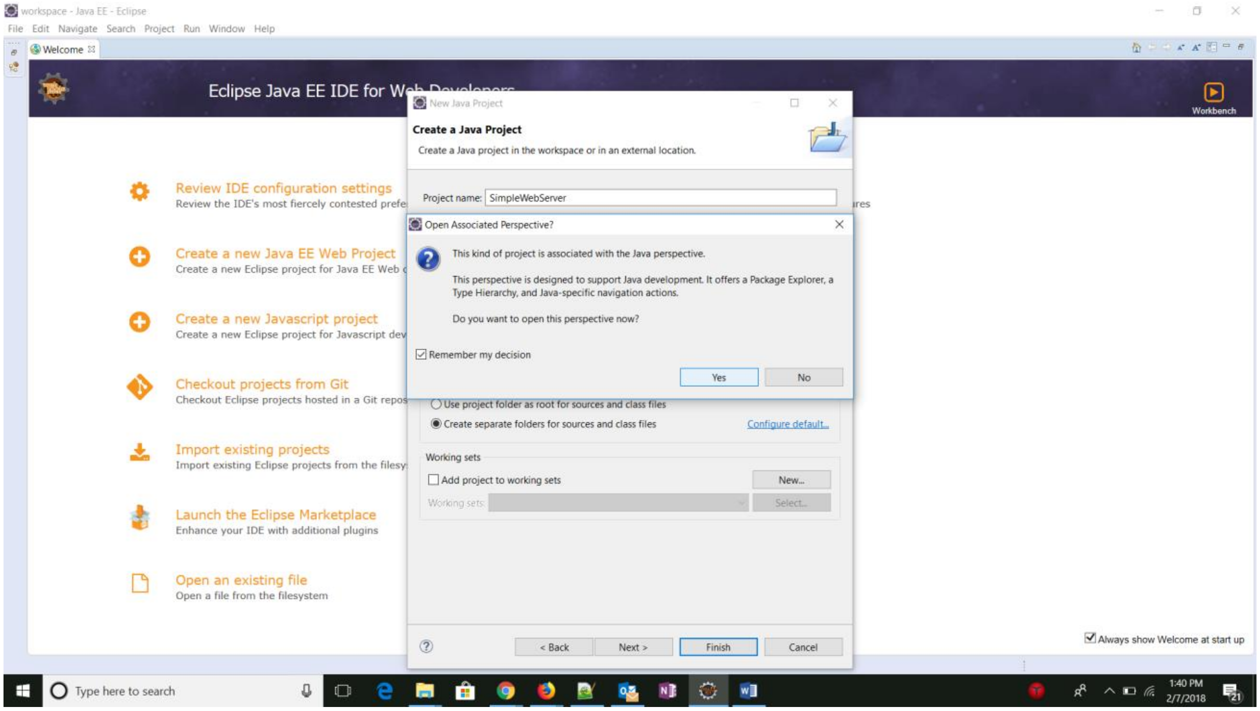
Working sets

☐ Add project to working sets

Working sets:

workspace - Java EE - Eclipse
File Edit Navigate Search Project Run Window Help
Welcome
Eclipse Java EE IDE for Web Developers
Workbench
Review IDE configuration settings
Create a new Java EE Web Project
Create a new Javascript project
Checkout projects from Git
Import existing projects
Launch the Eclipse Marketplace
Open an existing file
Browse...
Configure JREs...
Configure default...
New...
Select...
< Back
Next >
Finish
Cancel
Always show Welcome at start up
Type here to search
1:40 PM
2/7/2018

d.



Open Associated Perspective?

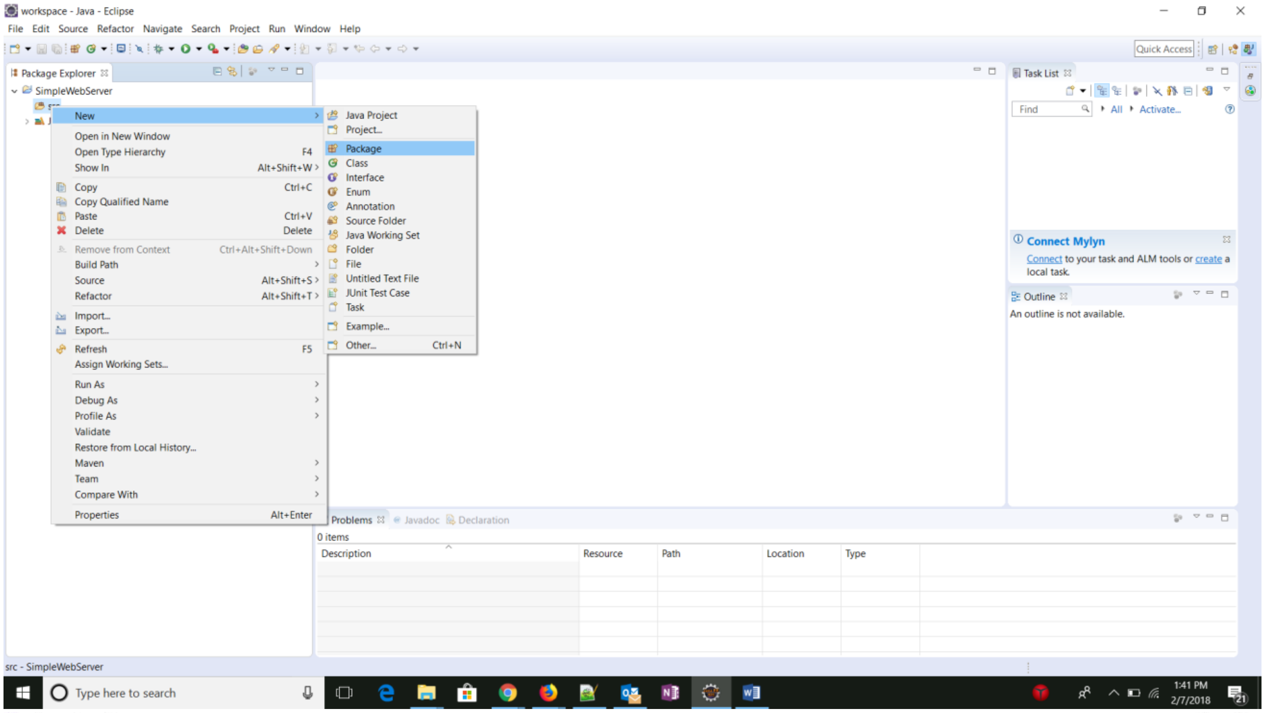
This kind of project is associated with the Java perspective.

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

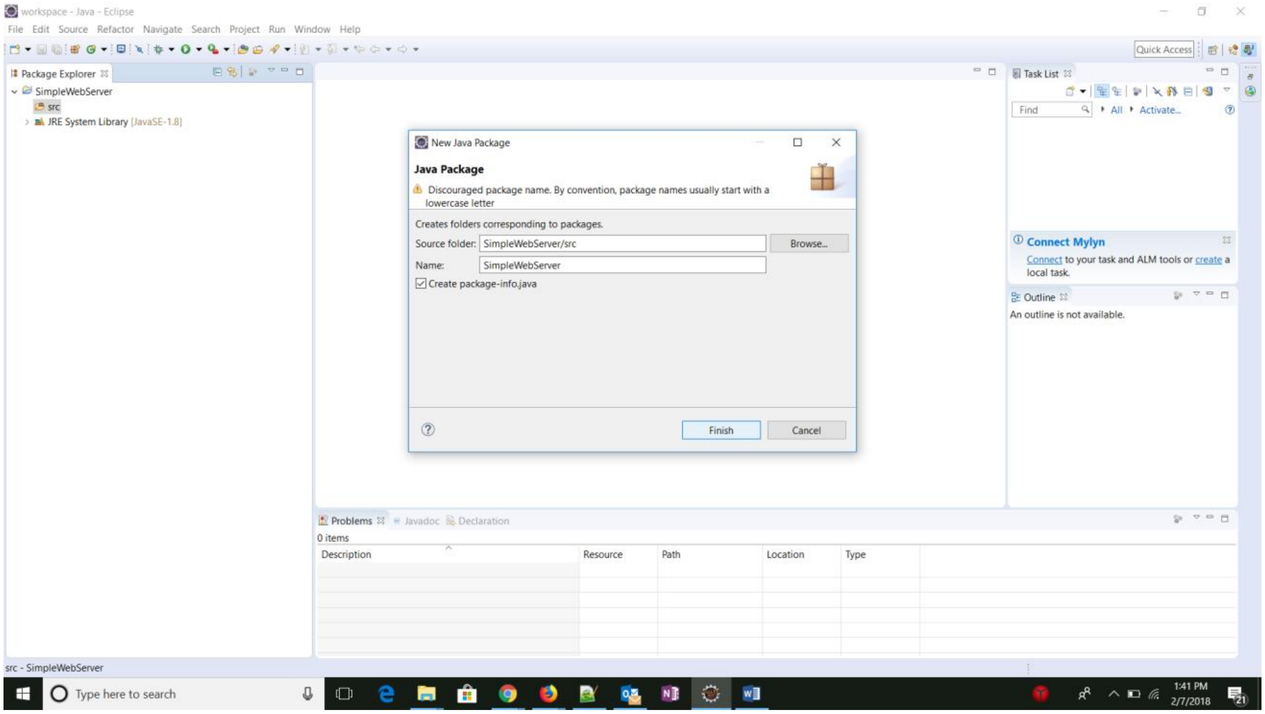
Do you want to open this perspective now?

☑ Remember my decision

e.



f.



Java Package

Discouraged package name. By convention, package names usually start with a lowercase letter

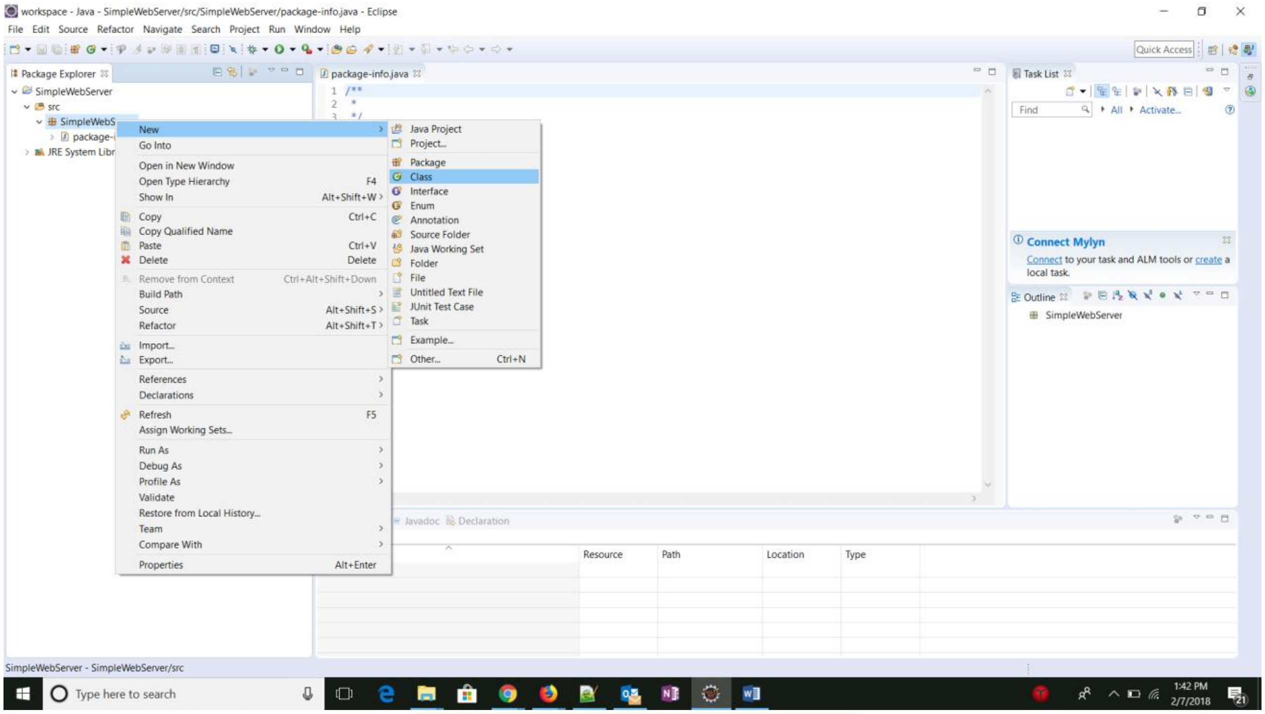
Creates folders corresponding to packages.

Source folder: SimpleWebServer/src

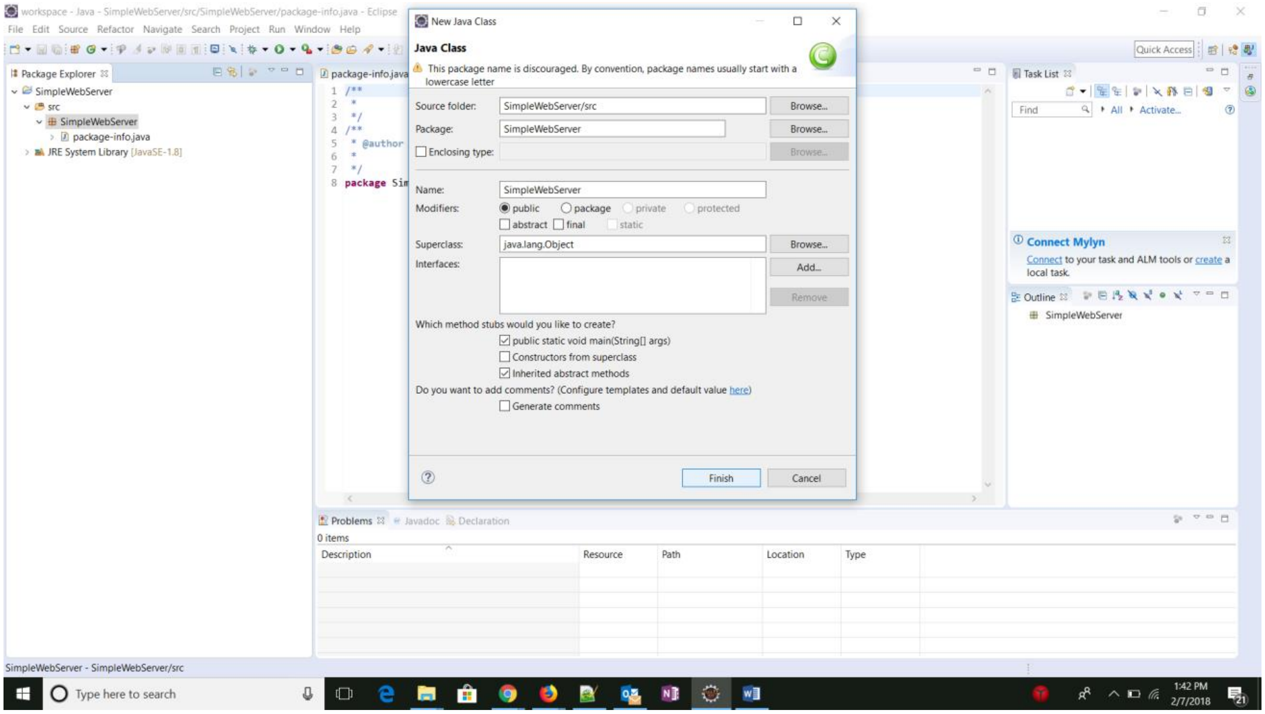
Name: SimpleWebServer

☑ Create package-info.java

g.



h.



```
1 /**
2  *
3  */
4 /**
5  * @author
6  *
7  */
8 package Sim
```

This package name is discouraged. By convention, package names usually start with a lowercase letter

Source folder: SimpleWebServer/src

Package: SimpleWebServer

□ Enclosing type:

Name: SimpleWebServer

Modifiers: ● public ○ package ○ private ○ protected
□ abstract □ final □ static

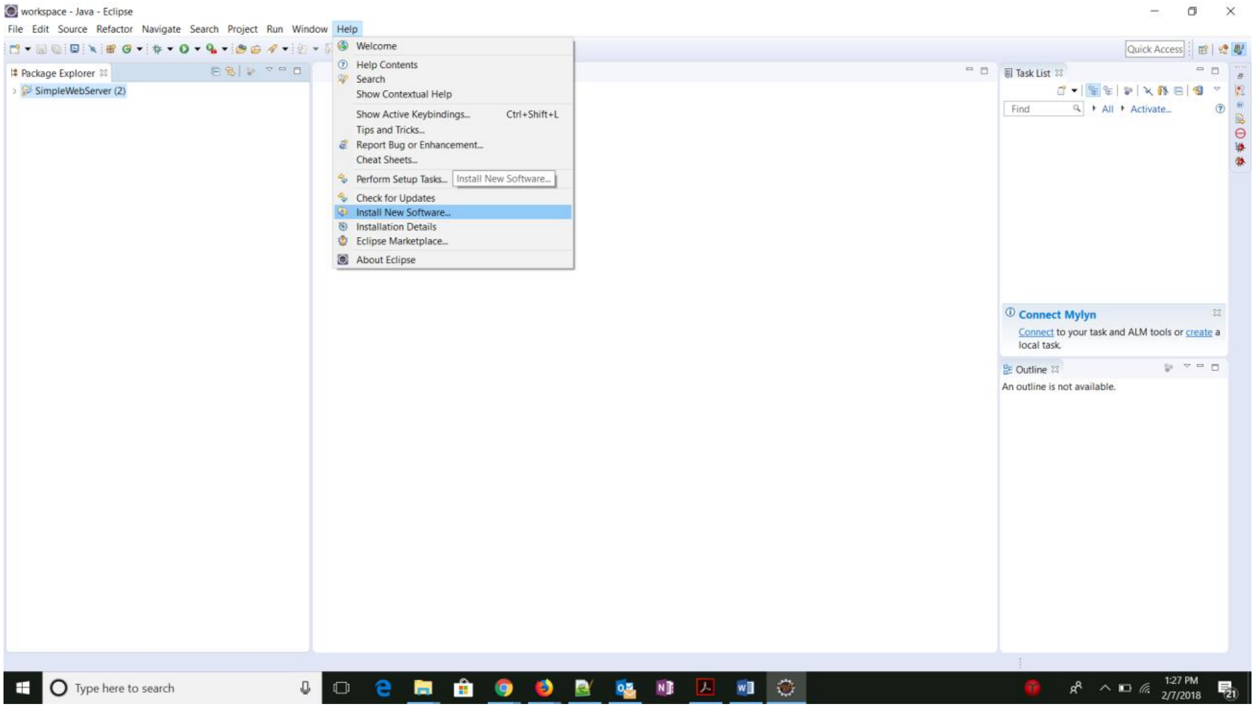
Superclass: java.lang.Object

Interfaces:

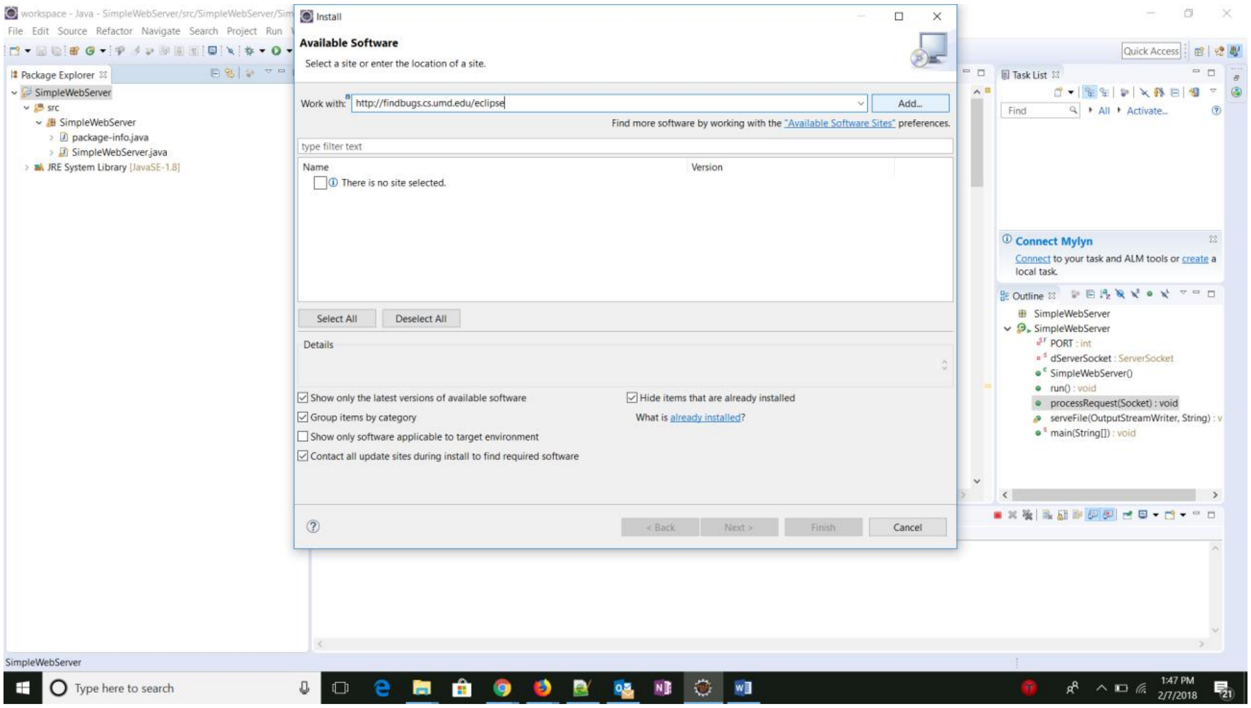
Which method stubs would you like to create?
☑ public static void main(String[] args)
□ Constructors from superclass
☑ Inherited abstract methods

Do you want to add comments? (Configure templates and default value here)
□ Generate comments

3. From Help tab, go to “Install New Software” to add the Findbugs plugin for eclipse as shown below:
    a. Click on “Install New Software”



b. Give the url to get the findbugs plugin for eclipse as: http://findbugs.cs.umd.edu/eclipse and click on "Add"


Install

Available Software

Select a site or enter the location of a site.

Work with: http://findbugs.cs.umd.edu/eclipse

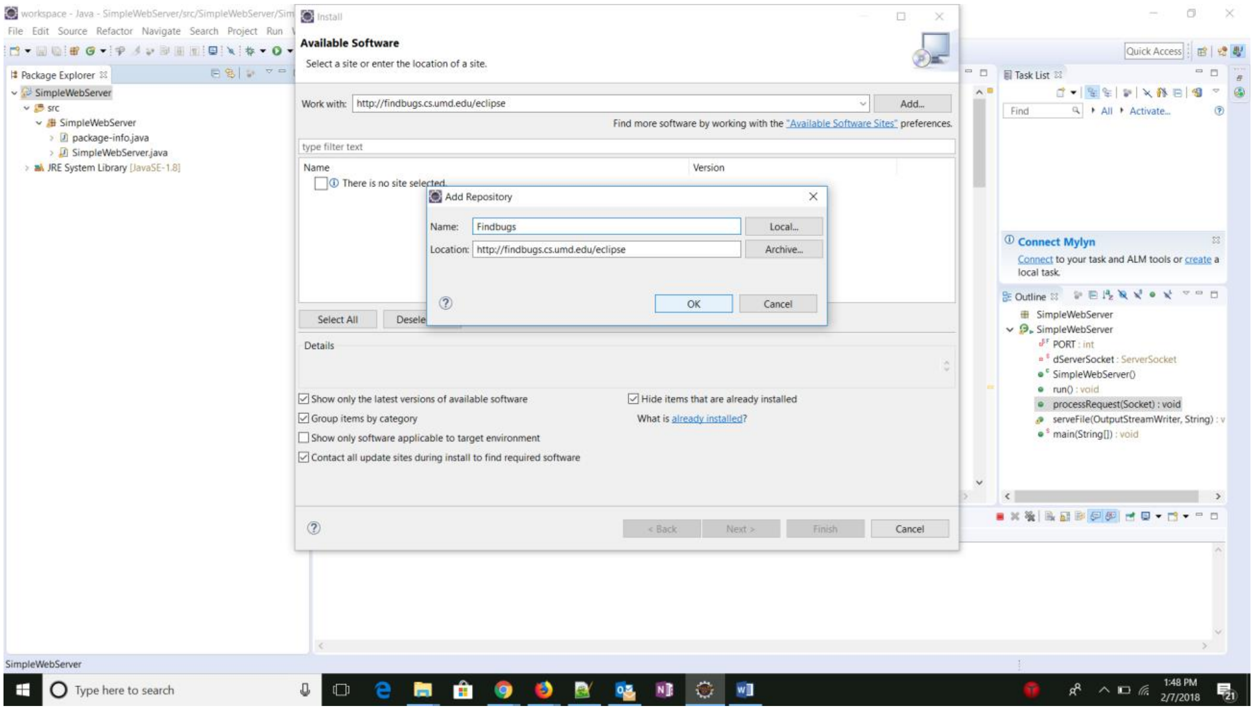
Find more software by working with the "Available Software Sites" preferences.

☐ There is no site selected.

☑ Show only the latest versions of available software
☑ Group items by category
☐ Show only software applicable to target environment
☑ Contact all update sites during install to find required software
☑ Hide items that are already installed

What is already installed?

c. The below window pops up, give the name as 'Findbugs' and click 'ok':


workspace - Java - SimpleWebServer/src/SimpleWebServer/Sim
File Edit Source Refactor Navigate Search Project Run
Package Explorer
SimpleWebServer
src
SimpleWebServer
package-info.java
SimpleWebServer.java
JRE System Library [JavaSE-1.8]
Quick Access
Task List
Find
All
Activate...
Connect Mylyn
Connect to your task and ALM tools or create a local task.
Outline
SimpleWebServer
SimpleWebServer
PORT : int
dServerSocket : ServerSocket
SimpleWebServer()
run() : void
processRequest(Socket) : void
serveFile(OutputStreamWriter, String) : v
main(String[]) : void
SimpleWebServer
Type here to search
1:48 PM
2/7/2018

Install

Available Software

Select a site or enter the location of a site.

Work with: http://findbugs.cs.umd.edu/eclipse

Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
□ ⓘ There is no site selected.	

Add Repository

Name: Findbugs

Location: http://findbugs.cs.umd.edu/eclipse

☑ Show only the latest versions of available software

☑ Group items by category

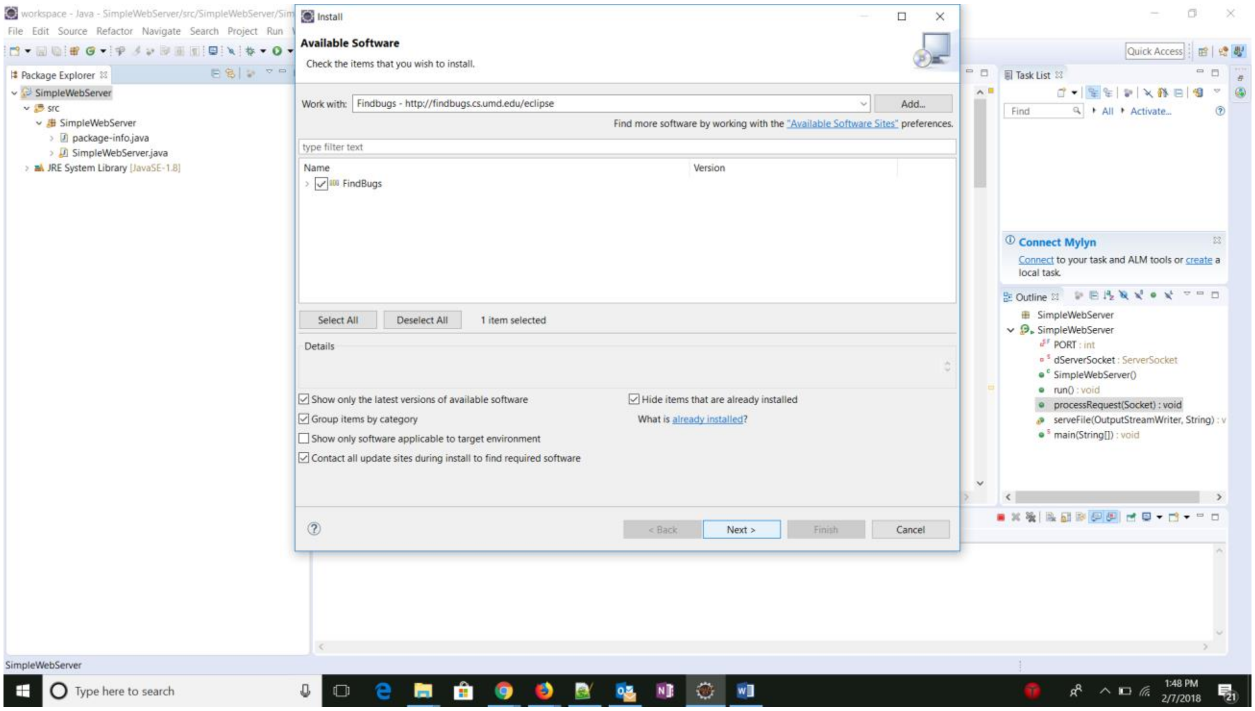
□ Show only software applicable to target environment

☑ Contact all update sites during install to find required software

☑ Hide items that are already installed

What is already installed?

d. Select ‘FindBugs’ as shown in the window below and click ‘Next’:


workspace - Java - SimpleWebServer/src/SimpleWebServer/Sim
File Edit Source Refactor Navigate Search Project Run
Package Explorer
SimpleWebServer
src
SimpleWebServer
package-info.java
SimpleWebServer.java
JRE System Library [JavaSE-1.8]
Quick Access
Task List
Find
All
Activate...
Connect Mylyn
Connect to your task and ALM tools or create a local task.
Outline
SimpleWebServer
SimpleWebServer
PORT : int
dServerSocket : ServerSocket
SimpleWebServer()
run() : void
processRequest(Socket) : void
serveFile(OutputStreamWriter, String) : v
main(String[]) : void
SimpleWebServer
Type here to search
1:48 PM
2/7/2018

Install

Available Software

Check the items that you wish to install.

Work with: Findbugs - http://findbugs.cs.umd.edu/eclipse

Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
☑ FindBugs	

1 item selected

☑ Show only the latest versions of available software

☑ Group items by category

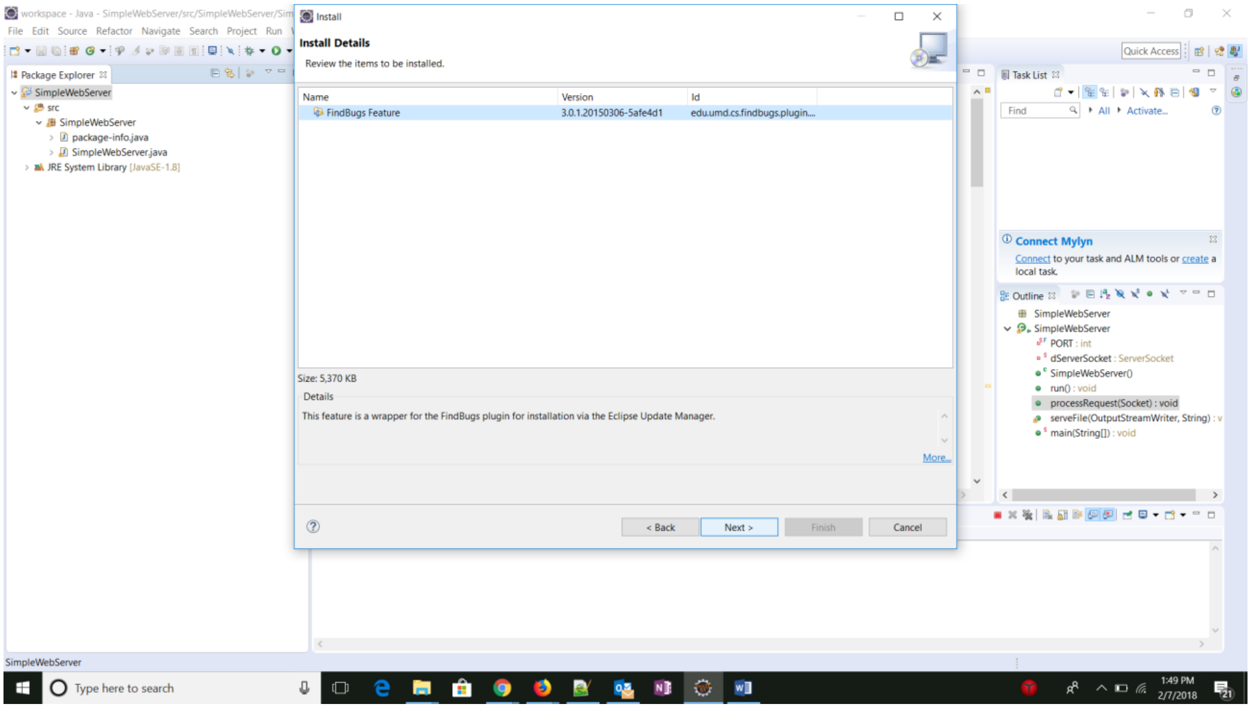
□ Show only software applicable to target environment

☑ Contact all update sites during install to find required software

☑ Hide items that are already installed

What is already installed?

e. Click ‘Next’ again as shown below:


Install

Install Details

Review the items to be installed.

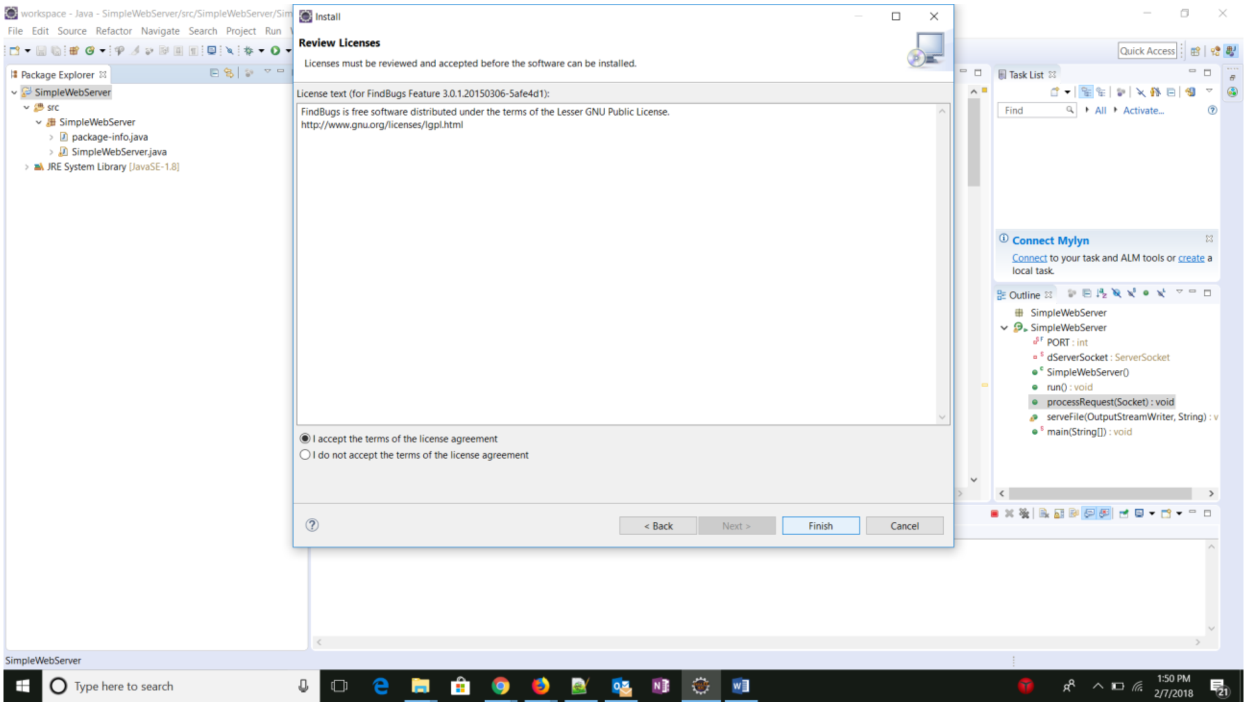
Name	Version	Id
FindBugs Feature	3.0.1.20150306-5afe4d1	edu.umd.cs.findbugs.plugin...

Size: 5,370 KB

Details

This feature is a wrapper for the FindBugs plugin for installation via the Eclipse Update Manager.

f. Select ‘I accept’ and click Finish to complete the installation Findbugs plugin for eclipse:


Install

Review Licenses

Licenses must be reviewed and accepted before the software can be installed.

License text (for FindBugs Feature 3.0.1.20150306-5afe4d1):

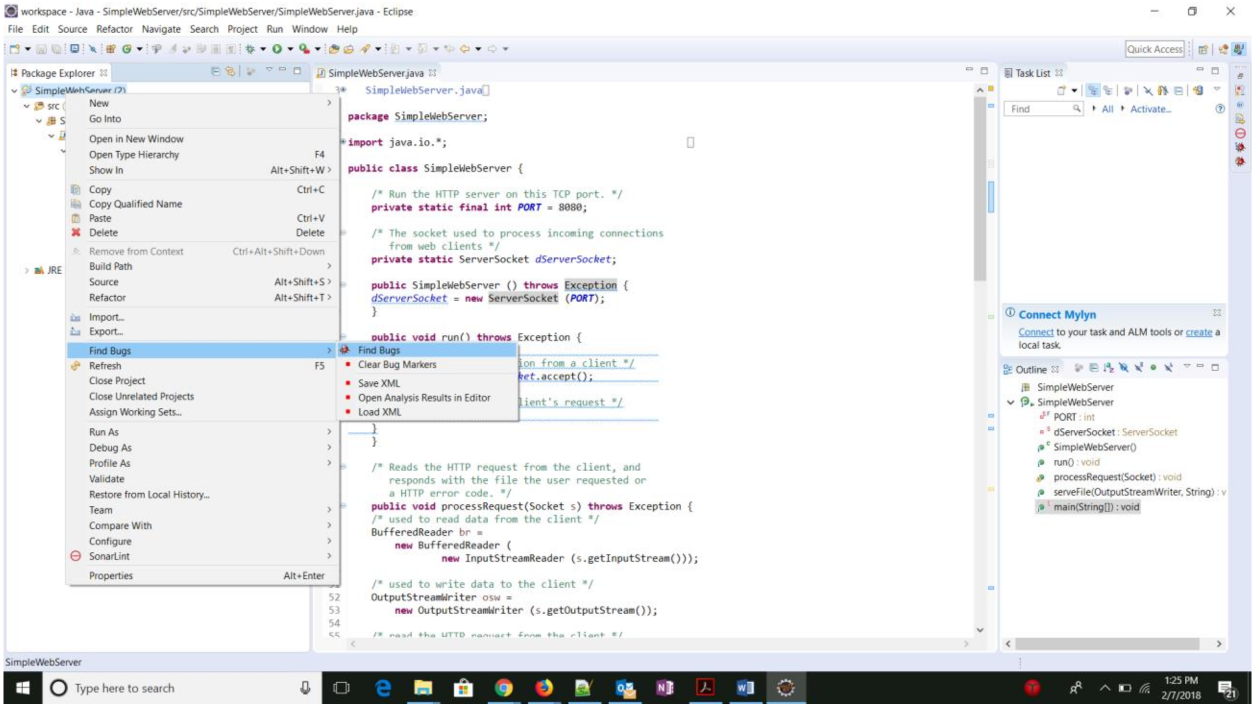
FindBugs is free software distributed under the terms of the Lesser GNU Public License.
http://www.gnu.org/licenses/lgpl.html

● I accept the terms of the license agreement
○ I do not accept the terms of the license agreement

g.

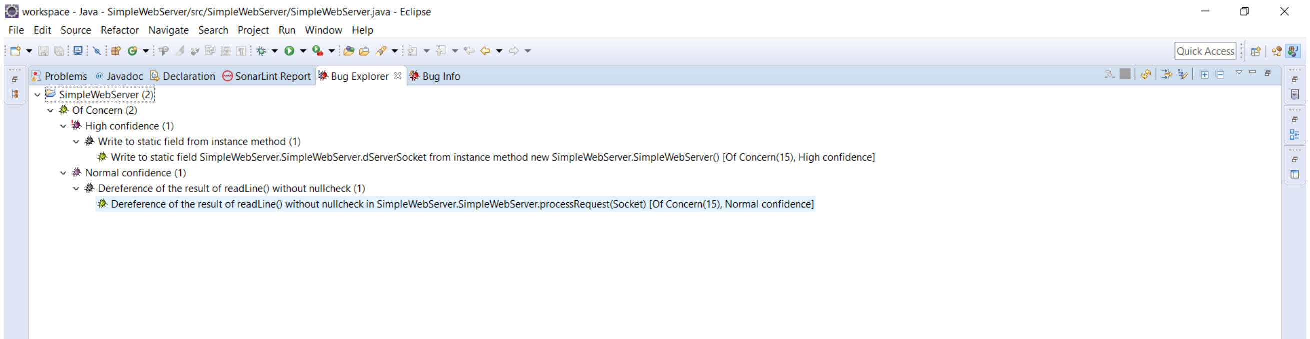
h. It will ask you to restart eclipse, click ok to restart eclipse.

i. To run the tool, select the project in the package explorer and select Find Bugs and in the sub menu, select ‘Find Bugs’ again:



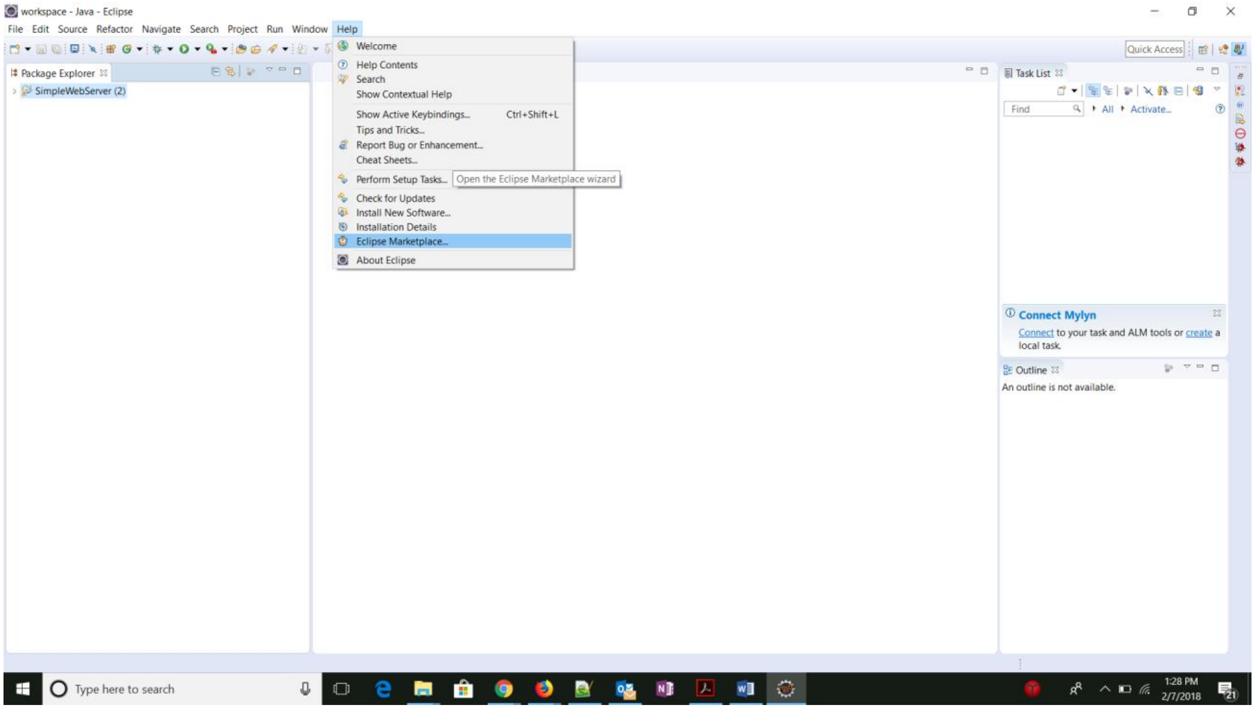
```
    SimpleWebServer.java

package SimpleWebServer;

import java.io.*;

public class SimpleWebServer {

    /* Run the HTTP server on this TCP port. */
    private static final int PORT = 8080;

    /* The socket used to process incoming connections
       from web clients */
    private static ServerSocket dServerSocket;

    public SimpleWebServer () throws Exception {
    dServerSocket = new ServerSocket (PORT);
    }

    public void run() throws Exception {
                              ion from a client */
                              ket.accept();

                              lient's request */

    }
    }

    /* Reads the HTTP request from the client, and
       responds with the file the user requested or
       a HTTP error code. */
    public void processRequest(Socket s) throws Exception {
    /* used to read data from the client */
    BufferedReader br =
        new BufferedReader (
            new InputStreamReader (s.getInputStream()));

    /* used to write data to the client */
    OutputStreamWriter osw =
        new OutputStreamWriter (s.getOutputStream());

    /* read the HTTP request from the client */
```

j. The below Find bugs explorer will display the bugs present in the code:


workspace - Java - SimpleWebServer/src/SimpleWebServer/SimpleWebServer.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Problems
Javadoc
Declaration
SonarLint Report
Bug Explorer
Bug Info

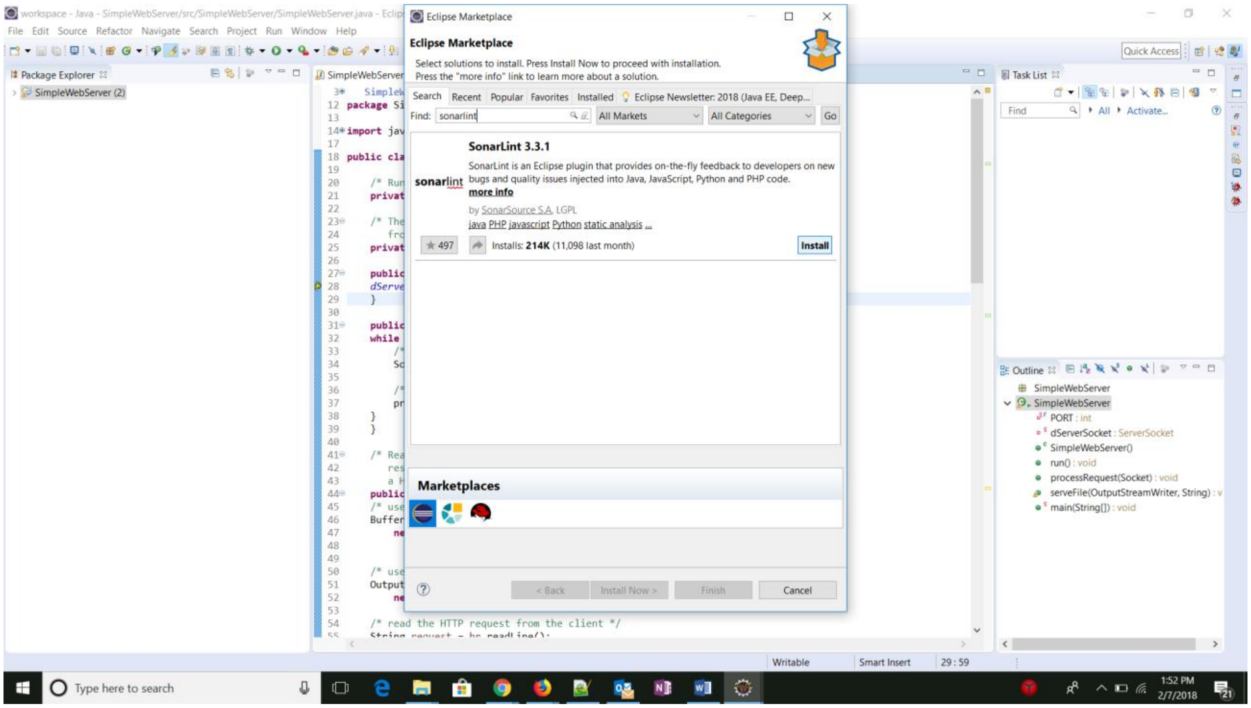
```
SimpleWebServer (2)
  Of Concern (2)
    High confidence (1)
      Write to static field from instance method (1)
        Write to static field SimpleWebServer.SimpleWebServer.dServerSocket from instance method new SimpleWebServer.SimpleWebServer() [Of Concern(15), High confidence]
    Normal confidence (1)
      Dereference of the result of readLine() without nullcheck (1)
        Dereference of the result of readLine() without nullcheck in SimpleWebServer.SimpleWebServer.processRequest(Socket) [Of Concern(15), Normal confidence]
```

4. To install SonarLint plugin for eclipse, go to “Eclipse Market place” from the Help tab as shown below:



a.

b. Type in “sonarlint” as shown in below window and click install to install the SonarLint plugin for eclipse:



Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.
Press the "more info" link to learn more about a solution.

Find: sonarlint — All Markets — All Categories

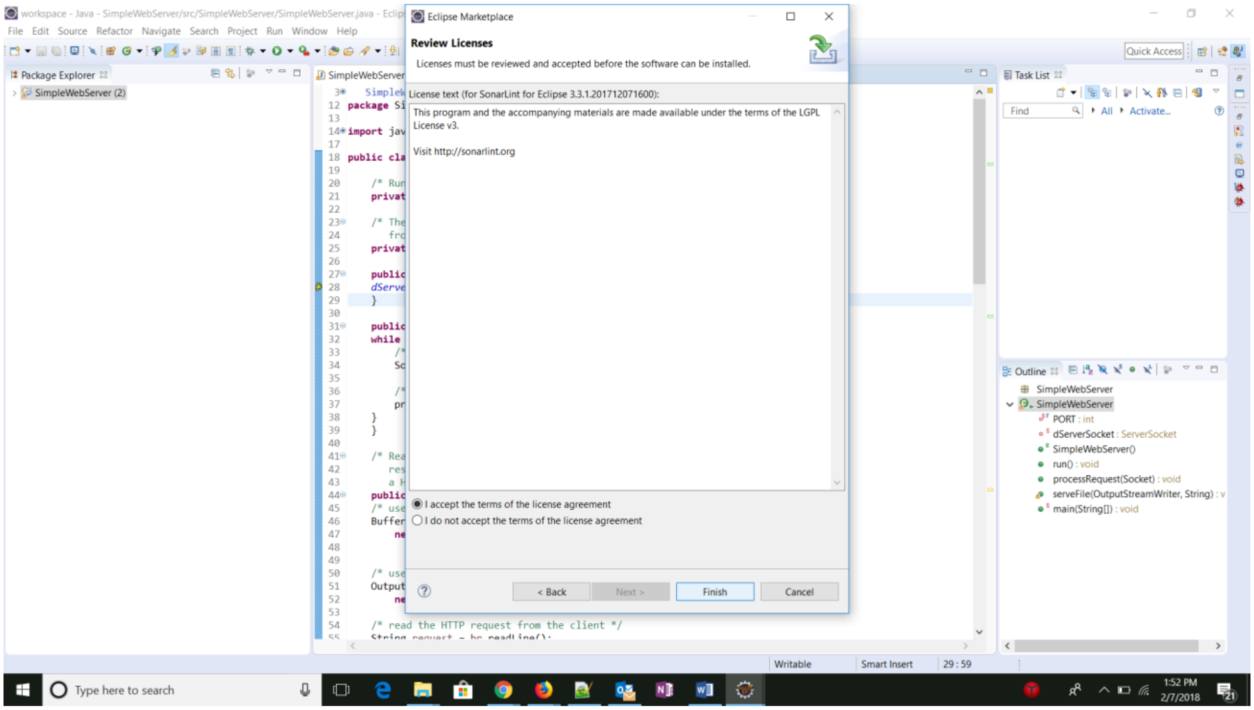
SonarLint 3.3.1

SonarLint is an Eclipse plugin that provides on-the-fly feedback to developers on new bugs and quality issues injected into Java, JavaScript, Python and PHP code.
more info

by SonarSource S.A, LGPL
java PHP javascript Python static analysis ...

★ 497 Installs: **214K** (11,098 last month)

c. Select ‘I Accept’ and click “Finish” to install the plugin as shown in below window:



Review Licenses

Licenses must be reviewed and accepted before the software can be installed.

License text (for SonarLint for Eclipse 3.3.1.201712071600):

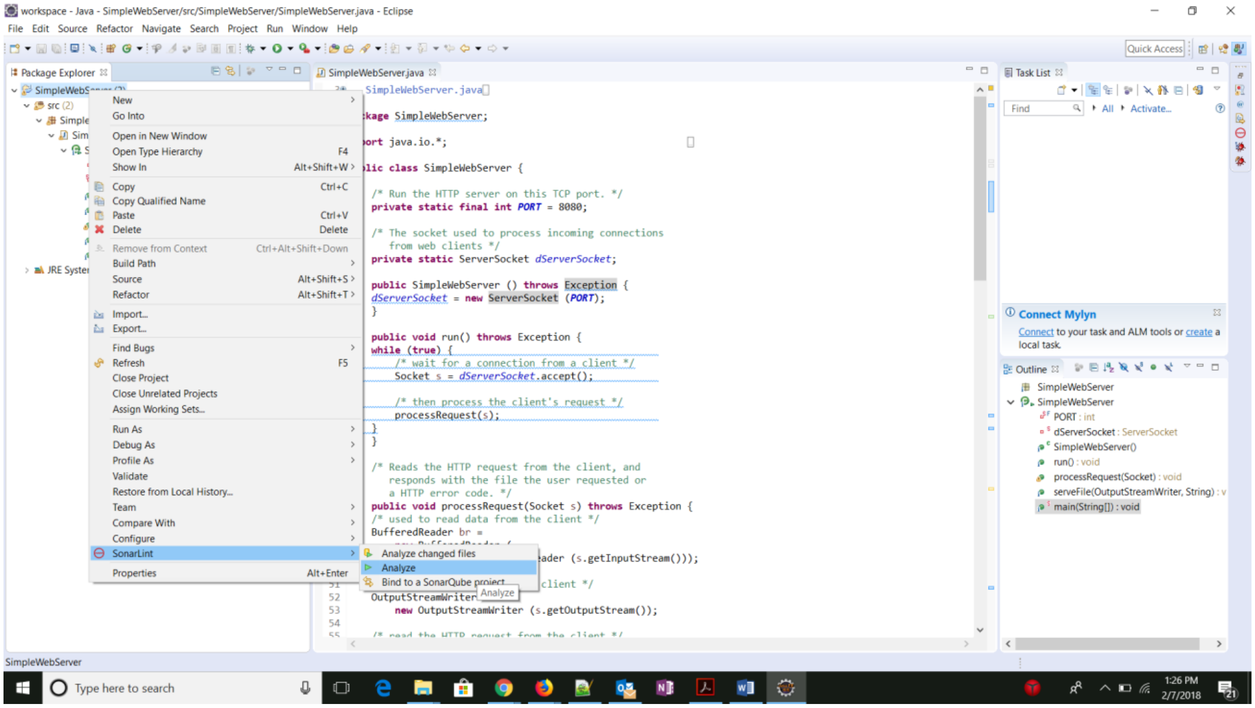
This program and the accompanying materials are made available under the terms of the LGPL License v3.

Visit http://sonarlint.org

● I accept the terms of the license agreement
○ I do not accept the terms of the license agreement

d. It will ask you to restart eclipse, click ok to restart eclipse.

e. To run the tool, select the project in the package explorer and select SonarLint, and in the sub menu, select ‘Analyze’ as shown in below window:



```
 SimpleWebServer.java

kage SimpleWebServer;

ort java.io.*;

lic class SimpleWebServer {

 /* Run the HTTP server on this TCP port. */
 private static final int PORT = 8080;

 /* The socket used to process incoming connections
    from web clients */
 private static ServerSocket dServerSocket;

 public SimpleWebServer () throws Exception {
 dServerSocket = new ServerSocket (PORT);
 }

 public void run() throws Exception {
 while (true) {
     /* wait for a connection from a client */
     Socket s = dServerSocket.accept();

     /* then process the client's request */
     processRequest(s);
 }
 }

 /* Reads the HTTP request from the client, and
    responds with the file the user requested or
    a HTTP error code. */
 public void processRequest(Socket s) throws Exception {
 /* used to read data from the client */
 BufferedReader br =
                                 ader (s.getInputStream()));

                                 client */
 OutputStreamWriter
     new OutputStreamWriter (s.getOutputStream());

 /* read the HTTP request from the client */
```

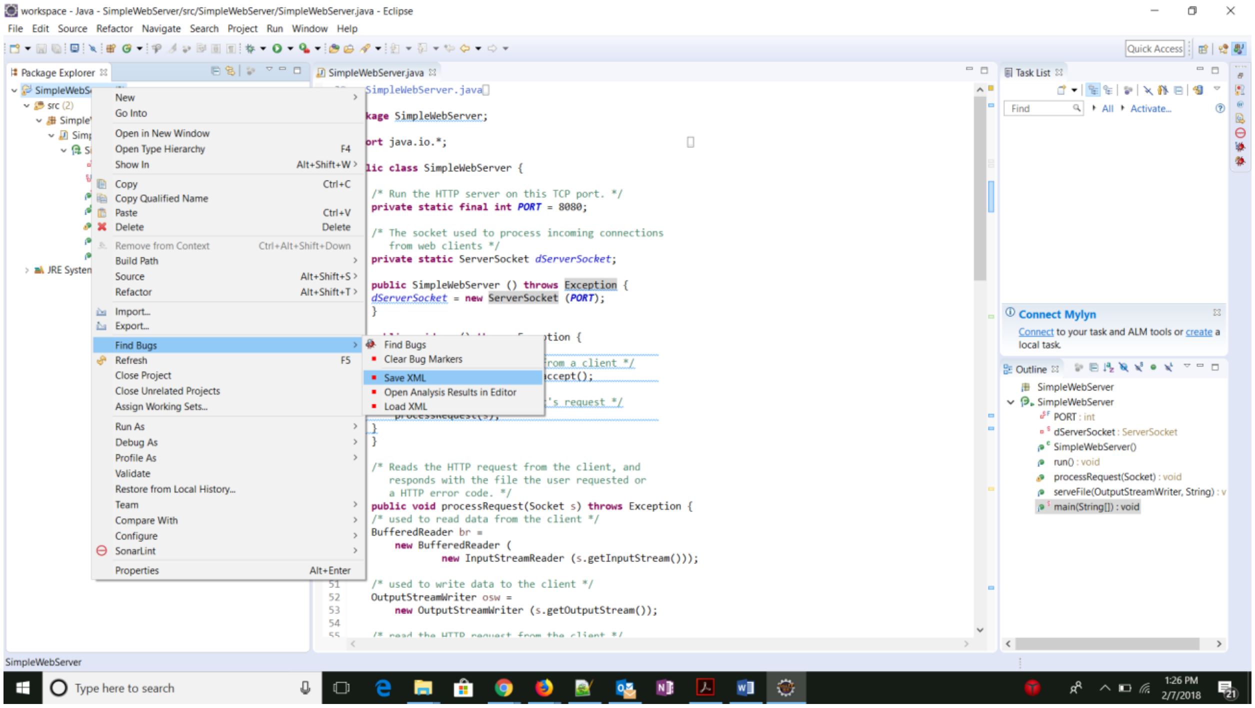
Comparison/ Contrast Tools

1. Does the tool analyze source or binary as input?
   - ✓Findbugs tool analyzes byte code which is why it is more prone to reporting false positives.
   - ✓SonarLint analyzes source code and provides much detailed information about the bugs it finds with a possible solution to fix the bug.
2. Which category of tools is it?
   a. Type checking
   b. Style checking
   c. Program understanding
   d. Program verification
   e. Property checking
   f. Bug finding
   g. Security review
   - ✓ Both the Findbugs and SonarLint are Bug finding tools.
3. Show an example (if one exists) of a finding that is reported by one tool and not others.
   - ✓In the method run(), end condition of the while loop is missing. This bug is reported by SonarLint tool but not by Findbugs tool.
4. Show an example (if one exists) of a finding reported by multiple tools.
   - ✓In Line 28, dServerSocket object is a static object and it is initialized in the constructor of the class which is not an acceptable practice. This bug is reported by both tools, Findbugs and SonarLint.
5. For the known flaw in the code used, document which tools reported it (true negative) and which tools did not (false positive).
   - ✓SonarLint reported that exception handling is generic in all the methods except serveFile of the SimpleWebServer class, whereas Findbugs tool did not report this obvious flaw in the code.

Results

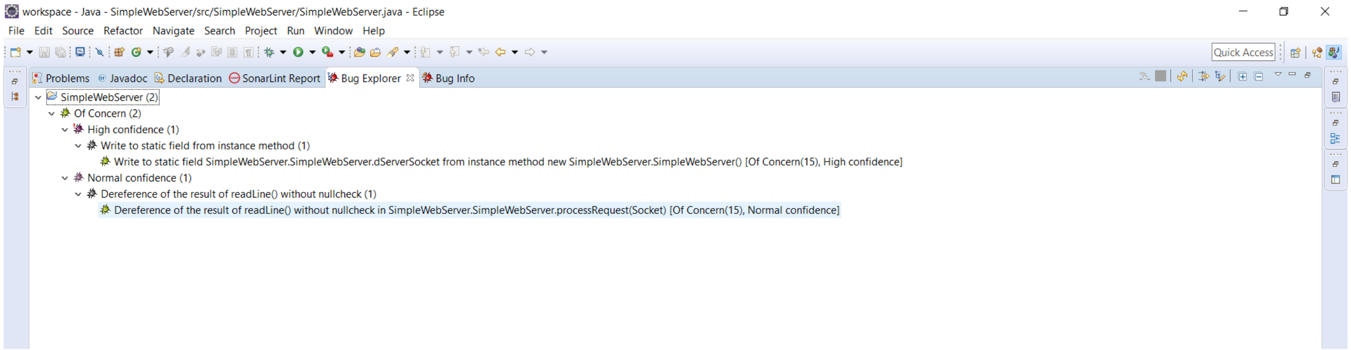
Findbugs:

- To save the bug report given by the Findbugs tool, select the project in the package explorer and select Find Bugs and in the sub menu, select ‘Save XML’ to save the report in XML format as shown below:



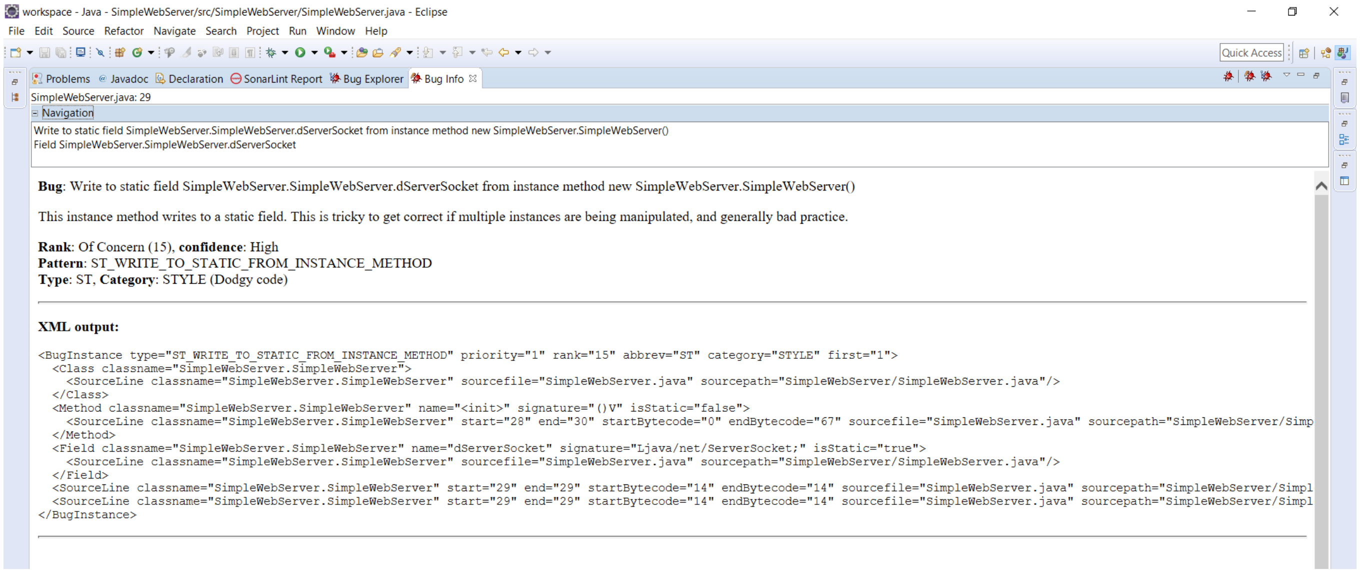
```
SimpleWebServer.java

kage SimpleWebServer;

ort java.io.*;

lic class SimpleWebServer {

 /* Run the HTTP server on this TCP port. */
 private static final int PORT = 8080;

 /* The socket used to process incoming connections
    from web clients */
 private static ServerSocket dServerSocket;

 public SimpleWebServer () throws Exception {
 dServerSocket = new ServerSocket (PORT);
 }

                       tion {

                       rom a client */
                       ccept();

                       's request */

 }
 }

 /* Reads the HTTP request from the client, and
    responds with the file the user requested or
    a HTTP error code. */
 public void processRequest(Socket s) throws Exception {
 /* used to read data from the client */
 BufferedReader br =
     new BufferedReader (
             new InputStreamReader (s.getInputStream()));

 /* used to write data to the client */
 OutputStreamWriter osw =
     new OutputStreamWriter (s.getOutputStream());
```

- Below are the screenshots of the bugs reported by the findbugs tool:


workspace - Java - SimpleWebServer/src/SimpleWebServer/SimpleWebServer.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Problems
Javadoc
Declaration
SonarLint Report
Bug Explorer
Bug Info

```
SimpleWebServer (2)
  Of Concern (2)
    High confidence (1)
      Write to static field from instance method (1)
        Write to static field SimpleWebServer.SimpleWebServer.dServerSocket from instance method new SimpleWebServer.SimpleWebServer() [Of Concern(15), High confidence]
    Normal confidence (1)
      Dereference of the result of readLine() without nullcheck (1)
        Dereference of the result of readLine() without nullcheck in SimpleWebServer.SimpleWebServer.processRequest(Socket) [Of Concern(15), Normal confidence]
```

(a)


workspace - Java - SimpleWebServer/src/SimpleWebServer/SimpleWebServer.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Problems
Javadoc
Declaration
SonarLint Report
Bug Explorer
Bug Info

SimpleWebServer.java: 29

Navigation

Write to static field SimpleWebServer.SimpleWebServer.dServerSocket from instance method new SimpleWebServer.SimpleWebServer()
Field SimpleWebServer.SimpleWebServer.dServerSocket

Bug: Write to static field SimpleWebServer.SimpleWebServer.dServerSocket from instance method new SimpleWebServer.SimpleWebServer()

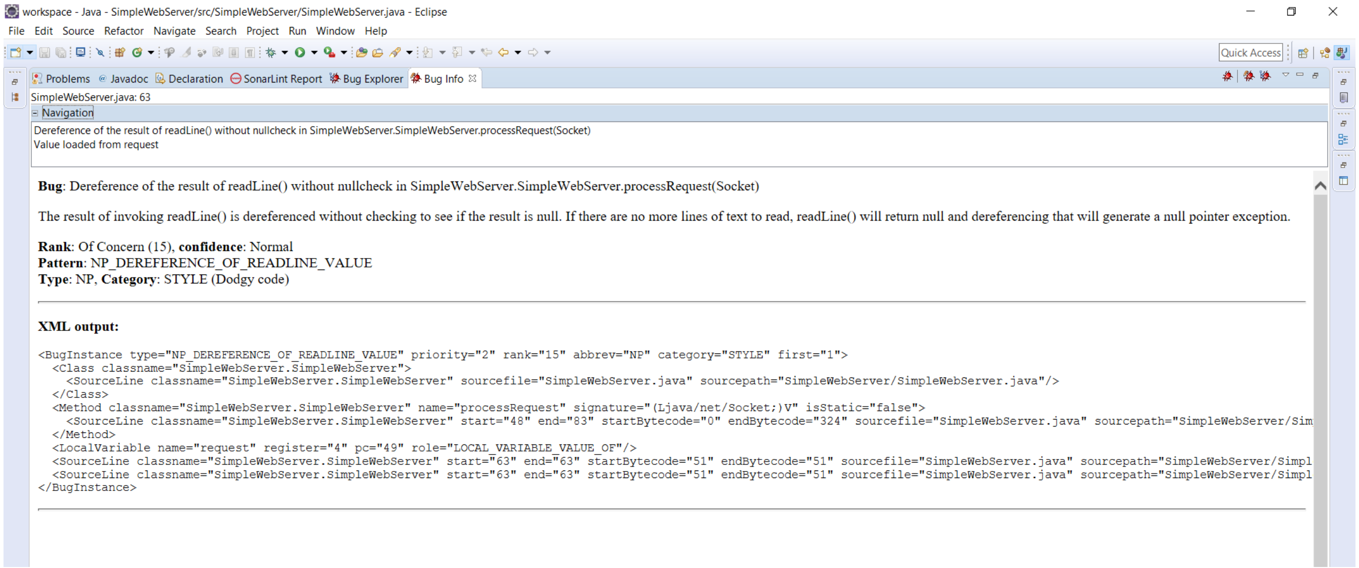
This instance method writes to a static field. This is tricky to get correct if multiple instances are being manipulated, and generally bad practice.

Rank: Of Concern (15), **confidence**: High
Pattern: ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD
Type: ST, **Category**: STYLE (Dodgy code)

XML output:

```
<BugInstance type="ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD" priority="1" rank="15" abbrev="ST" category="STYLE" first="1">
  <Class classname="SimpleWebServer.SimpleWebServer">
    <SourceLine classname="SimpleWebServer.SimpleWebServer" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/SimpleWebServer.java"/>
  </Class>
  <Method classname="SimpleWebServer.SimpleWebServer" name="<init>" signature="()V" isStatic="false">
    <SourceLine classname="SimpleWebServer.SimpleWebServer" start="28" end="30" startBytecode="0" endBytecode="67" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/Simp
  </Method>
  <Field classname="SimpleWebServer.SimpleWebServer" name="dServerSocket" signature="Ljava/net/ServerSocket;" isStatic="true">
    <SourceLine classname="SimpleWebServer.SimpleWebServer" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/SimpleWebServer.java"/>
  </Field>
  <SourceLine classname="SimpleWebServer.SimpleWebServer" start="29" end="29" startBytecode="14" endBytecode="14" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/Simpl
  <SourceLine classname="SimpleWebServer.SimpleWebServer" start="29" end="29" startBytecode="14" endBytecode="14" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/Simpl
</BugInstance>
```

(b)


workspace - Java - SimpleWebServer/src/SimpleWebServer/SimpleWebServer.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Problems
Javadoc
Declaration
SonarLint Report
Bug Explorer
Bug Info

SimpleWebServer.java: 63

Navigation

Dereference of the result of readLine() without nullcheck in SimpleWebServer.SimpleWebServer.processRequest(Socket)
Value loaded from request

Bug: Dereference of the result of readLine() without nullcheck in SimpleWebServer.SimpleWebServer.processRequest(Socket)

The result of invoking readLine() is dereferenced without checking to see if the result is null. If there are no more lines of text to read, readLine() will return null and dereferencing that will generate a null pointer exception.

Rank: Of Concern (15), **confidence**: Normal
Pattern: NP_DEREFERENCE_OF_READLINE_VALUE
Type: NP, **Category**: STYLE (Dodgy code)

XML output:

```
<BugInstance type="NP_DEREFERENCE_OF_READLINE_VALUE" priority="2" rank="15" abbrev="NP" category="STYLE" first="1">
  <Class classname="SimpleWebServer.SimpleWebServer">
    <SourceLine classname="SimpleWebServer.SimpleWebServer" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/SimpleWebServer.java"/>
  </Class>
  <Method classname="SimpleWebServer.SimpleWebServer" name="processRequest" signature="(Ljava/net/Socket;)V" isStatic="false">
    <SourceLine classname="SimpleWebServer.SimpleWebServer" start="48" end="83" startBytecode="0" endBytecode="324" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/Sim
  </Method>
  <LocalVariable name="request" register="4" pc="49" role="LOCAL_VARIABLE_VALUE_OF"/>
  <SourceLine classname="SimpleWebServer.SimpleWebServer" start="63" end="63" startBytecode="51" endBytecode="51" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/Simpl
  <SourceLine classname="SimpleWebServer.SimpleWebServer" start="63" end="63" startBytecode="51" endBytecode="51" sourcefile="SimpleWebServer.java" sourcepath="SimpleWebServer/Simpl
</BugInstance>
```

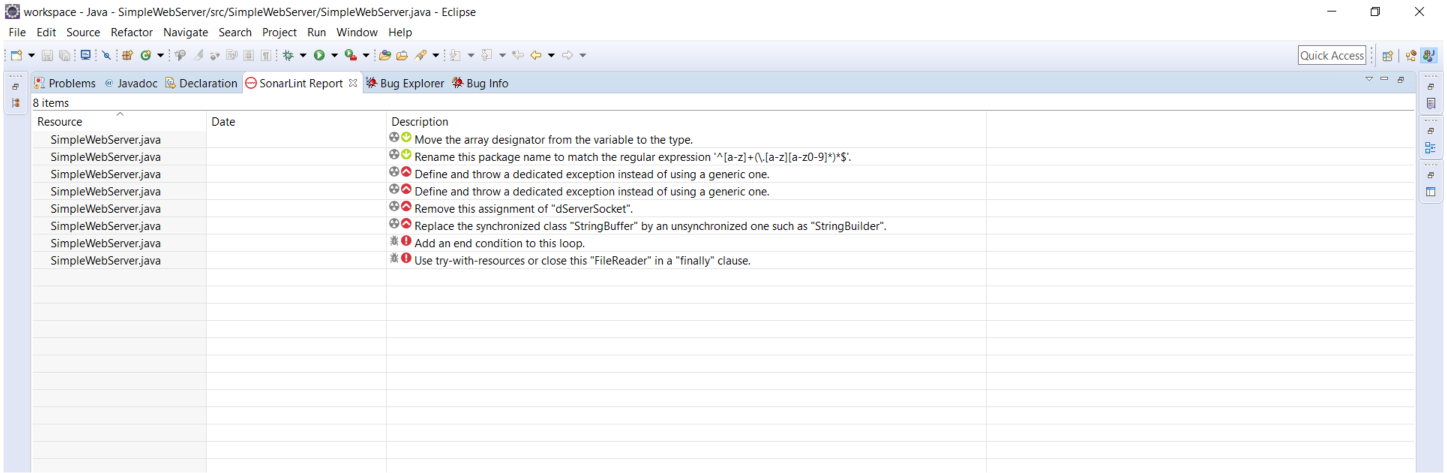
(c)

- Below is the XML report:

SimpleWebServer_Fin
dBugs_Report.xml

SonarLint:

Below is the screenshot of the SonarLint report:



workspace - Java - SimpleWebServer/src/SimpleWebServer/SimpleWebServer.java - Eclipse

8 items

Resource	Date	Description
SimpleWebServer.java		Move the array designator from the variable to the type.
SimpleWebServer.java		Rename this package name to match the regular expression '^[a-z]+(\.[a-z][a-z0-9]*)*$'.
SimpleWebServer.java		Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java		Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java		Remove this assignment of "dServerSocket".
SimpleWebServer.java		Replace the synchronized class "StringBuffer" by an unsynchronized one such as "StringBuilder".
SimpleWebServer.java		Add an end condition to this loop.
SimpleWebServer.java		Use try-with-resources or close this "FileReader" in a "finally" clause.

Part 2

Code Provided

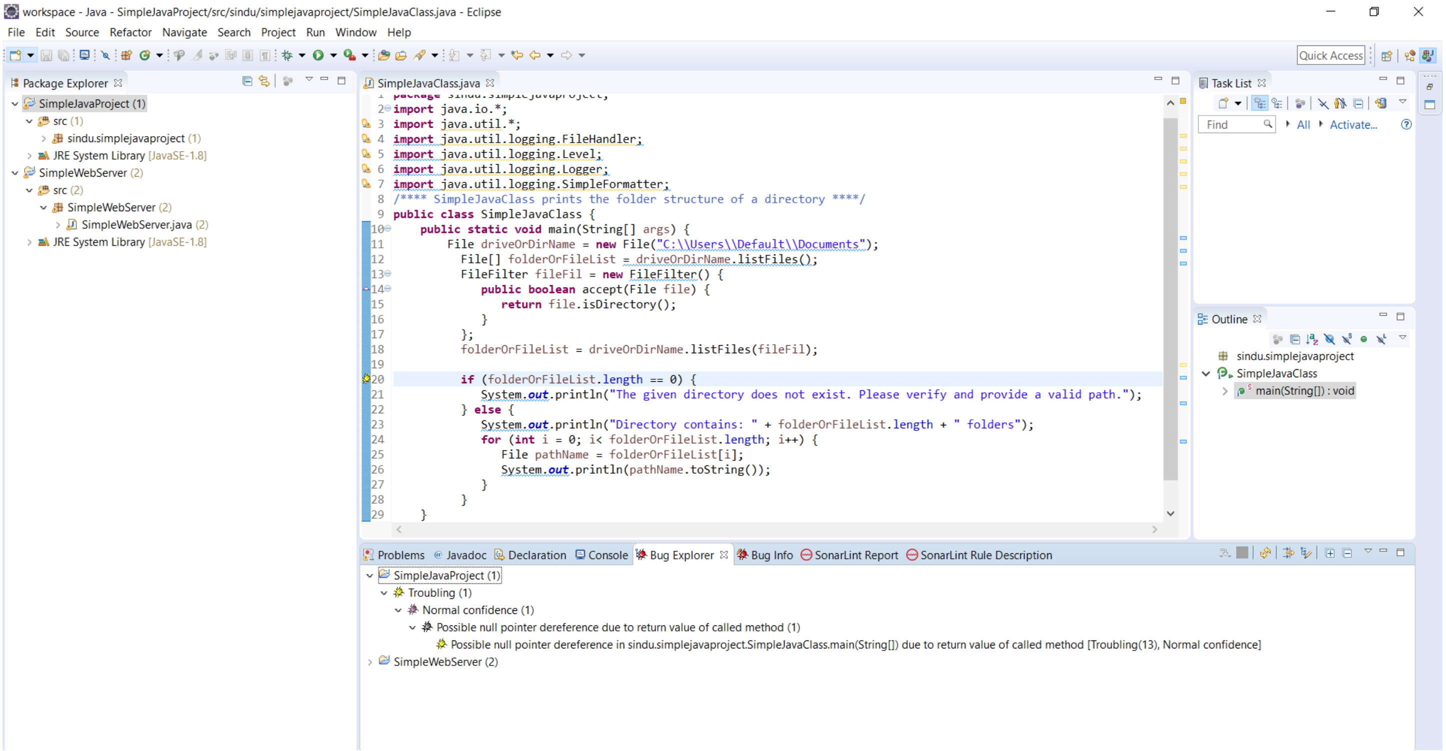
The code used for the part 2 of this project, returns the folder structure/ directory of a drive or a path.

Initial_SimpleJavaClass.java

Results

Below are the results returned by Findbugs and SonarLint tools:

Findbugs:



workspace - Java - SimpleJavaProject/src/sindu/simplejavaproject/SimpleJavaClass.java - Eclipse

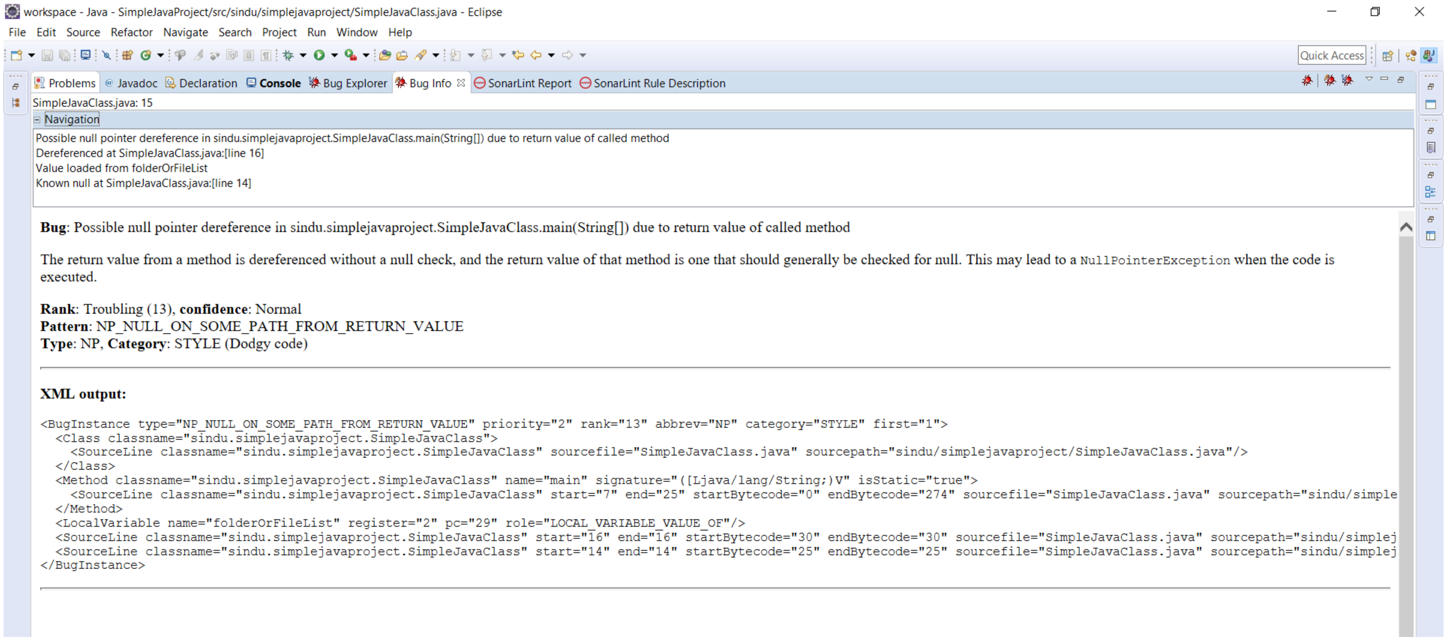
```
SimpleJavaProject (1)
  src (1)
    sindu.simplejavaproject (1)
  JRE System Library [JavaSE-1.8]
SimpleWebServer (2)
  src (2)
    SimpleWebServer (2)
      SimpleWebServer.java (2)
  JRE System Library [JavaSE-1.8]
```

```java
package sindu.simplejavaproject;
import java.io.*;
import java.util.*;
import java.util.logging.FileHandler;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.util.logging.SimpleFormatter;
/**** SimpleJavaClass prints the folder structure of a directory ****/
public class SimpleJavaClass {
    public static void main(String[] args) {
        File driveOrDirName = new File("C:\\Users\\Default\\Documents");
          File[] folderOrFileList = driveOrDirName.listFiles();
          FileFilter fileFil = new FileFilter() {
             public boolean accept(File file) {
                return file.isDirectory();
             }
          };
          folderOrFileList = driveOrDirName.listFiles(fileFil);

          if (folderOrFileList.length == 0) {
             System.out.println("The given directory does not exist. Please verify and provide a valid path.");
          } else {
             System.out.println("Directory contains: " + folderOrFileList.length + " folders");
             for (int i = 0; i< folderOrFileList.length; i++) {
                File pathName = folderOrFileList[i];
                System.out.println(pathName.toString());
             }
          }
    }
```

```
sindu.simplejavaproject
SimpleJavaClass
  main(String[]) : void
```

```
SimpleJavaProject (1)
  Troubling (1)
    Normal confidence (1)
      Possible null pointer dereference due to return value of called method (1)
        Possible null pointer dereference in sindu.simplejavaproject.SimpleJavaClass.main(String[]) due to return value of called method [Troubling(13), Normal confidence]
SimpleWebServer (2)
```

(a)


workspace - Java - SimpleJavaProject/src/sindu/simplejavaproject/SimpleJavaClass.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Problems
Javadoc
Declaration
Console
Bug Explorer
Bug Info
SonarLint Report
SonarLint Rule Description
SimpleJavaClass.java: 15
Navigation

Possible null pointer dereference in sindu.simplejavaproject.SimpleJavaClass.main(String[]) due to return value of called method
Dereferenced at SimpleJavaClass.java:[line 16]
Value loaded from folderOrFileList
Known null at SimpleJavaClass.java:[line 14]

Bug: Possible null pointer dereference in sindu.simplejavaproject.SimpleJavaClass.main(String[]) due to return value of called method

The return value from a method is dereferenced without a null check, and the return value of that method is one that should generally be checked for null. This may lead to a `NullPointerException` when the code is executed.

Rank: Troubling (13), **confidence**: Normal
Pattern: NP_NULL_ON_SOME_PATH_FROM_RETURN_VALUE
Type: NP, **Category**: STYLE (Dodgy code)

XML output:

```
<BugInstance type="NP_NULL_ON_SOME_PATH_FROM_RETURN_VALUE" priority="2" rank="13" abbrev="NP" category="STYLE" first="1">
  <Class classname="sindu.simplejavaproject.SimpleJavaClass">
    <SourceLine classname="sindu.simplejavaproject.SimpleJavaClass" sourcefile="SimpleJavaClass.java" sourcepath="sindu/simplejavaproject/SimpleJavaClass.java"/>
  </Class>
  <Method classname="sindu.simplejavaproject.SimpleJavaClass" name="main" signature="([Ljava/lang/String;)V" isStatic="true">
    <SourceLine classname="sindu.simplejavaproject.SimpleJavaClass" start="7" end="25" startBytecode="0" endBytecode="274" sourcefile="SimpleJavaClass.java" sourcepath="sindu/simple
  </Method>
  <LocalVariable name="folderOrFileList" register="2" pc="29" role="LOCAL_VARIABLE_VALUE_OF"/>
  <SourceLine classname="sindu.simplejavaproject.SimpleJavaClass" start="16" end="16" startBytecode="30" endBytecode="30" sourcefile="SimpleJavaClass.java" sourcepath="sindu/simplej
  <SourceLine classname="sindu.simplejavaproject.SimpleJavaClass" start="14" end="14" startBytecode="25" endBytecode="25" sourcefile="SimpleJavaClass.java" sourcepath="sindu/simplej
</BugInstance>
```

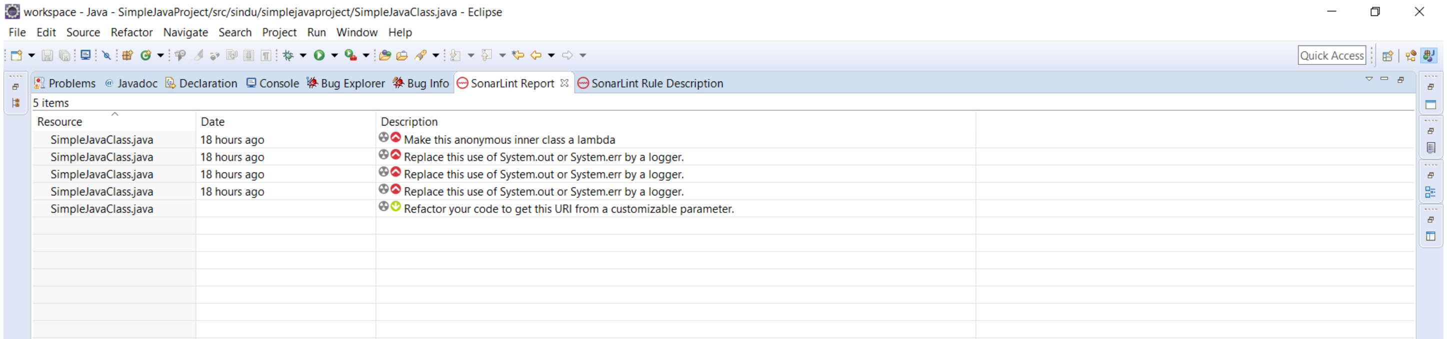
(b)

Below is the XML report:

SimpleJavaProject_Findbugs_Report.xml

SonarLint:

Below is the screenshot of the SonarLint report:


workspace - Java - SimpleJavaProject/src/sindu/simplejavaproject/SimpleJavaClass.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Problems
Javadoc
Declaration
Console
Bug Explorer
Bug Info
SonarLint Report
SonarLint Rule Description
5 items

Resource	Date	Description
SimpleJavaClass.java	18 hours ago	Make this anonymous inner class a lambda
SimpleJavaClass.java	18 hours ago	Replace this use of System.out or System.err by a logger.
SimpleJavaClass.java	18 hours ago	Replace this use of System.out or System.err by a logger.
SimpleJavaClass.java	18 hours ago	Replace this use of System.out or System.err by a logger.
SimpleJavaClass.java		Refactor your code to get this URI from a customizable parameter.

Fixes

Code after bug fixes:

SimpleJavaClass.java

Findbugs: 1 Normal Confidence bug - Fixed

"Possible null pointer dereference in sindu.simplejavaproject.SimpleJavaClass.main(String[]) due to return value of called method [Troubling(13), Normal confidence]"

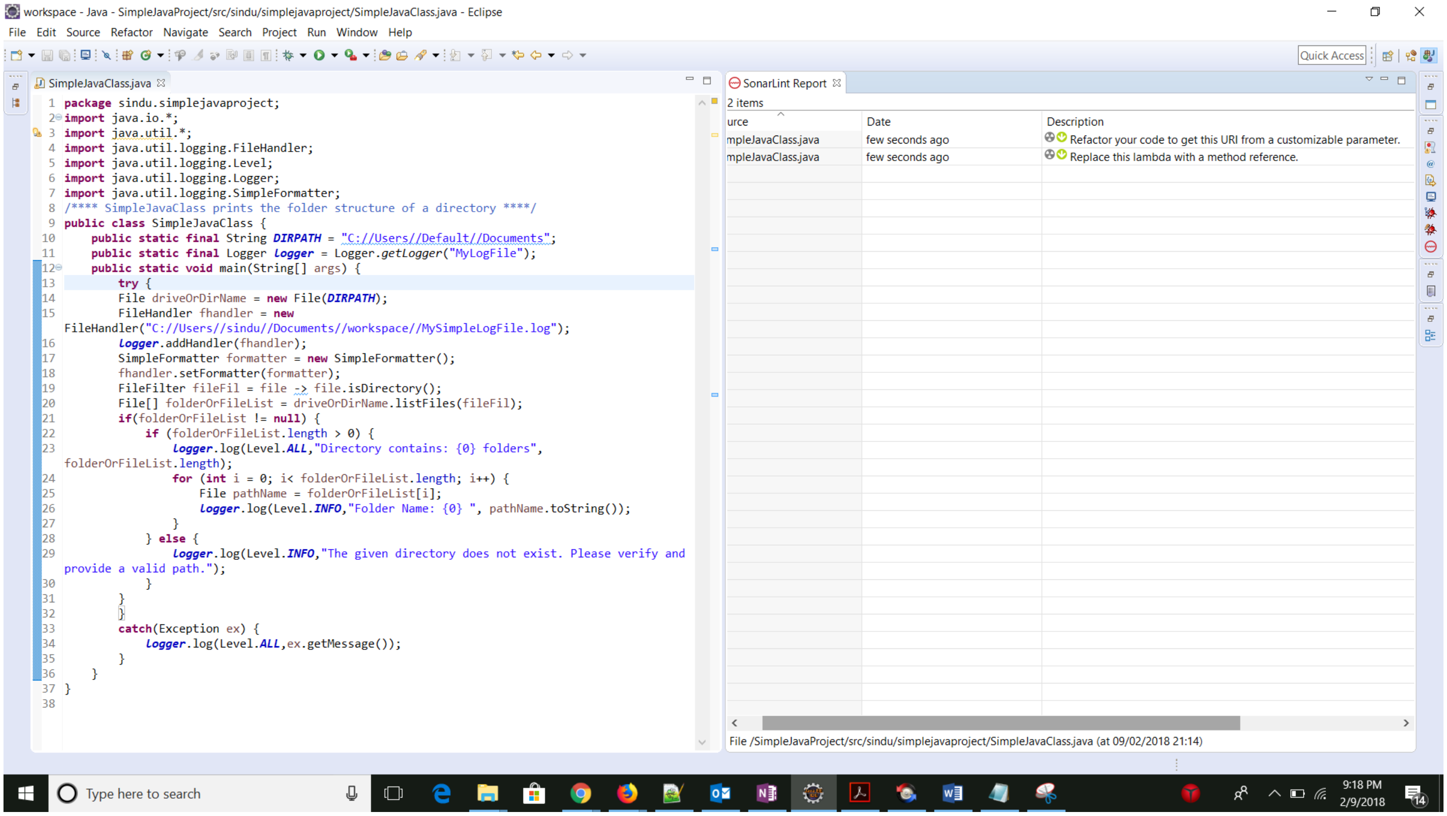
As shown in the fig(a) in above section, the bug reported by the tool means that, on line 20, null check was not implemented for the folderOrFileList object. Once null check was added, no bugs were reported in the bug explorer.

SonarLint: 4 major bugs, 1 minor bug

Fixed:

3 of the major bugs as shown in the sonarlint report above are about using logger class instead writing the logs to console. Another major bug is about converting an inner class to lambda expression.

All the major bugs are fixed and the new report now has two minor bugs:


workspace - Java - SimpleJavaProject/src/sindu/simplejavaproject/SimpleJavaClass.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
SimpleJavaClass.java
SonarLint Report
Type here to search
9:18 PM
2/9/2018

```
package sindu.simplejavaproject;
import java.io.*;
import java.util.*;
import java.util.logging.FileHandler;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.util.logging.SimpleFormatter;
/**** SimpleJavaClass prints the folder structure of a directory ****/
public class SimpleJavaClass {
    public static final String DIRPATH = "C://Users//Default//Documents";
    public static final Logger logger = Logger.getLogger("MyLogFile");
    public static void main(String[] args) {
        try {
        File driveOrDirName = new File(DIRPATH);
        FileHandler fhandler = new
FileHandler("C://Users//sindu//Documents//workspace//MySimpleLogFile.log");
        logger.addHandler(fhandler);
        SimpleFormatter formatter = new SimpleFormatter();
        fhandler.setFormatter(formatter);
        FileFilter fileFil = file -> file.isDirectory();
        File[] folderOrFileList = driveOrDirName.listFiles(fileFil);
        if(folderOrFileList != null) {
            if (folderOrFileList.length > 0) {
                logger.log(Level.ALL,"Directory contains: {0} folders",
folderOrFileList.length);
                for (int i = 0; i< folderOrFileList.length; i++) {
                    File pathName = folderOrFileList[i];
                    logger.log(Level.INFO,"Folder Name: {0} ", pathName.toString());
                }
            } else {
                logger.log(Level.INFO,"The given directory does not exist. Please verify and
provide a valid path.");
            }
        }
        }
        catch(Exception ex) {
            logger.log(Level.ALL,ex.getMessage());
        }
    }
}
```

2 items

urce	Date	Description
mpleJavaClass.java	few seconds ago	Refactor your code to get this URI from a customizable parameter.
mpleJavaClass.java	few seconds ago	Replace this lambda with a method reference.

File /SimpleJavaProject/src/sindu/simplejavaproject/SimpleJavaClass.java (at 09/02/2018 21:14)