
SDKs - Eclipse

SENG 403, Tutorial 2

AGENDA

- SDK Basics
 - Eclipse
 - How to create Project
 - How to create a Class
 - Run Program
 - Debug Program
-

SDK Basics

Software Development Kit is a set of software development tools that allows developers to create applications for a certain software package, software framework, hardware platform, computer system, video game console, operating system, or similar development platform.

- It often presents debugging tools and other utilities in an integrated development environment (IDE).
 - SDKs often include sample code and documentation as technical support.
-

SDK Basics

Integrated Development Environment – allows the automation of many of the common programming tasks in one environment

- Writing the code
 - Checking for Syntax (Language) errors
 - Compiling and Interpreting(Transferring to computer language)
 - Debugging (Fixing Run-time or Logic Errors)
 - Running the Application
-

Eclipse

What is Eclipse?

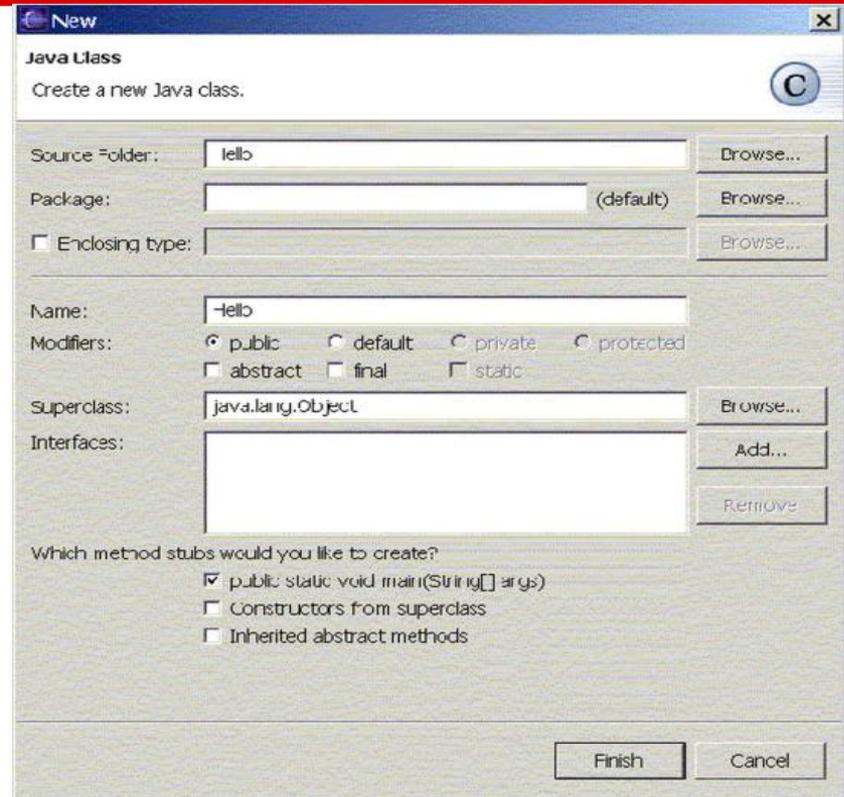
Eclipse is a Java-based, extensible open source development platform. By itself, it is simply a framework and a set of services for building a development environment from plug-in components. Fortunately, Eclipse comes with a standard set of plug-ins, including the Java Development Tools (JDT).

Eclipse

Creating a New Class

Using the Java perspective, right-click on the Hello project, and select **New > Class**. In the dialog box that appears, type "Hello" as the class name. Under "Which method stubs would you like to create?" check "public static void main(String[] args)," then **Finish**.

This will create a .java file with a Hello class and an empty main() method in the editor area



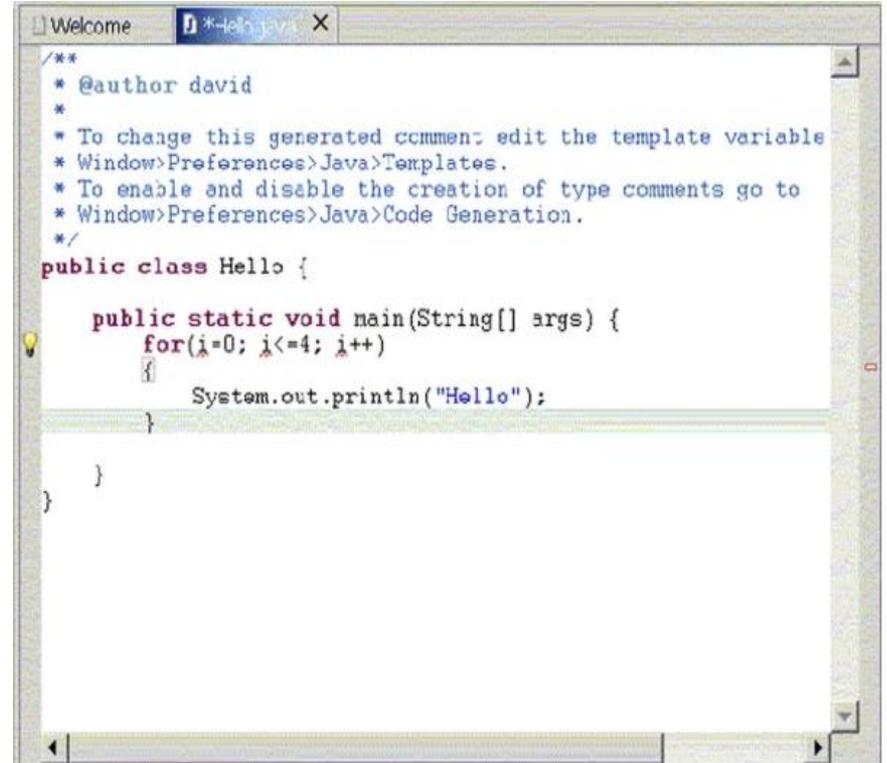
Eclipse

Hello Class

You'll notice some of the Eclipse editor's features as you type, including syntax checking and code completion.

When you type an open parenthesis or double quote, Eclipse will provide its partner automatically and place the cursor inside the pair.

In other cases, you can invoke code completion by using **Ctrl+Space**. Code completion provides a context-sensitive list of suggestions selectable by keyboard or mouse.



```
Welcome | *-1eh.java X
/**
 * @author david
 *
 * To change this generated comment: edit the template variable
 * Window>Preferences>Java>Templates.
 * To enable and disable the creation of type comments go to
 * Window>Preferences>Java>Code Generation.
 */
public class Hello {

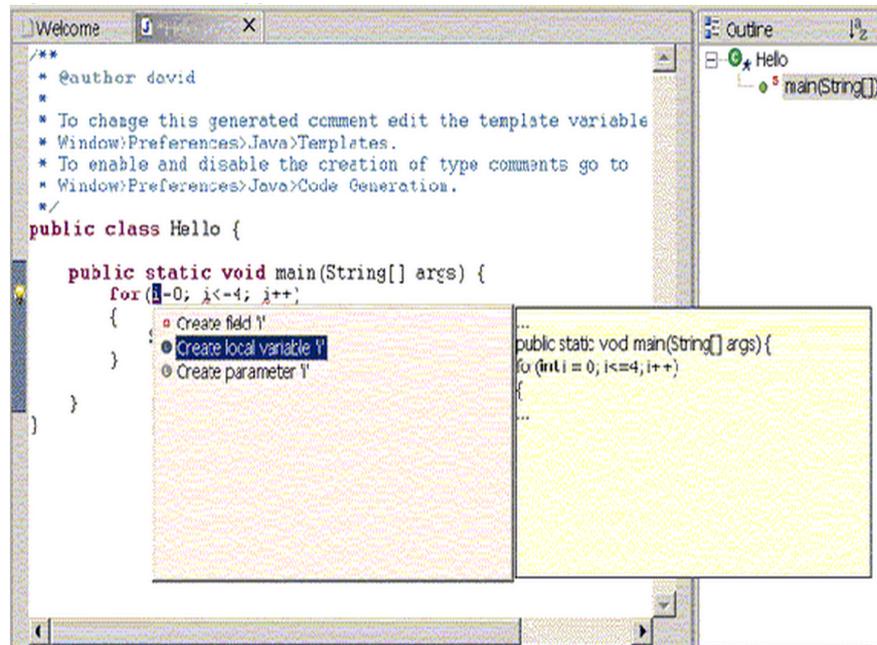
    public static void main(String[] args) {
        for(i=0; i<=4; i++)
        {
            System.out.println("Hello");
        }
    }
}
```

Eclipse

Hello Class - Error checking

Notice the light bulb or error icon next to the for statement. Double-clicking on the icon will bring up a list of suggested fixes.

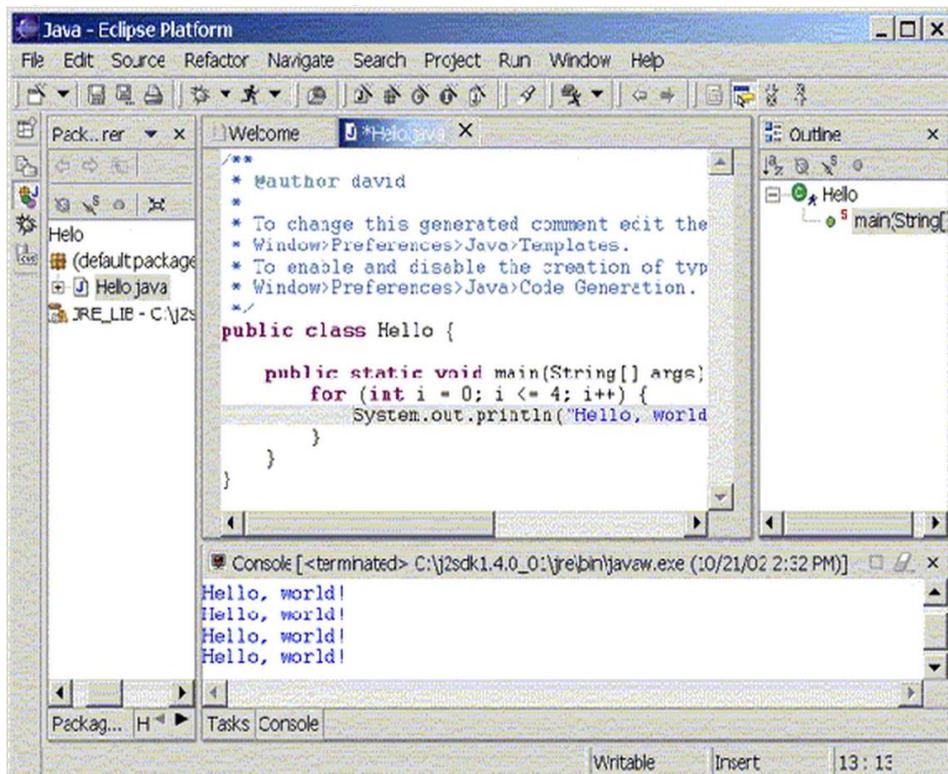
In this case, it will offer to create a class field `i`, a local variable `i`, or a method parameter `i`; clicking on each of these suggestions will display the code that would be generated.



Eclipse

Running the program

Once the code compiles without error, you can execute the program by selecting **Run** from the Eclipse menu. Run the program with the default settings. Notice that a new tabbed panel appears in the lower panel (the Console), displaying the program's output

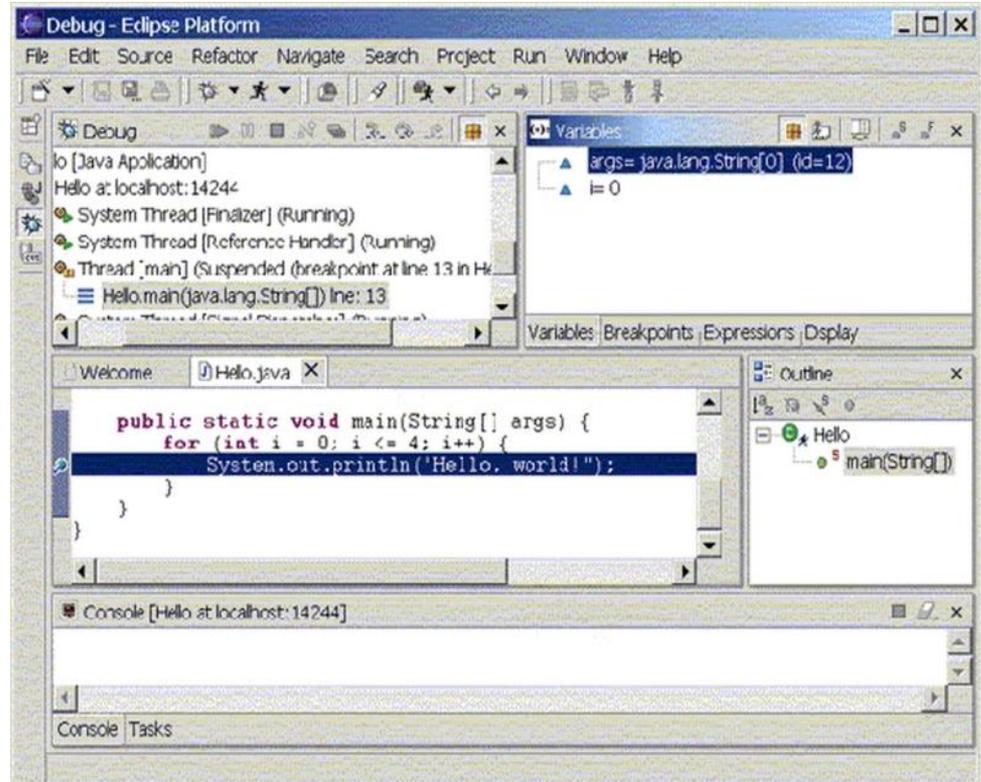


Eclipse

Debugging the program

Set a breakpoint in `main()` `System.out.println()` by double-clicking in the gray margin on the left side of the editor view, next to the call to `System.out.println()`. A blue dot will appear.

From the **Run** menu, select **Debug** and run the program with the default configurations. The perspective will automatically change to the Debug perspective

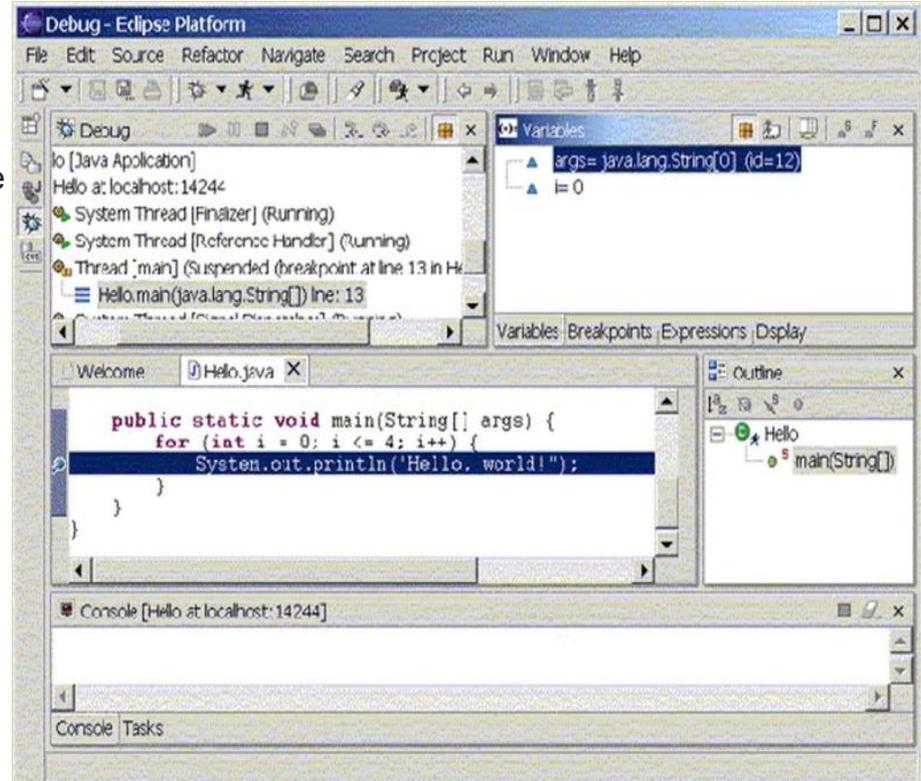


Eclipse

Debugging the program

Debug view at the top left of the perspective: shows the call stack and has a toolbar in the title bar that allows you to control the execution of the program (resume, suspend, terminate, step into the next statement, step over the next statement or return from a method).

The panel at the top right contains a number of tabbed views, including Variables, Breakpoints, Expressions, and Display.



References & Examples

[Download Eclipse](#)

[Getting Started With Eclipse](#)

[Recommended Plugins for Eclipse](#)
