PHAS3459
Scientific programming using object-oriented languages

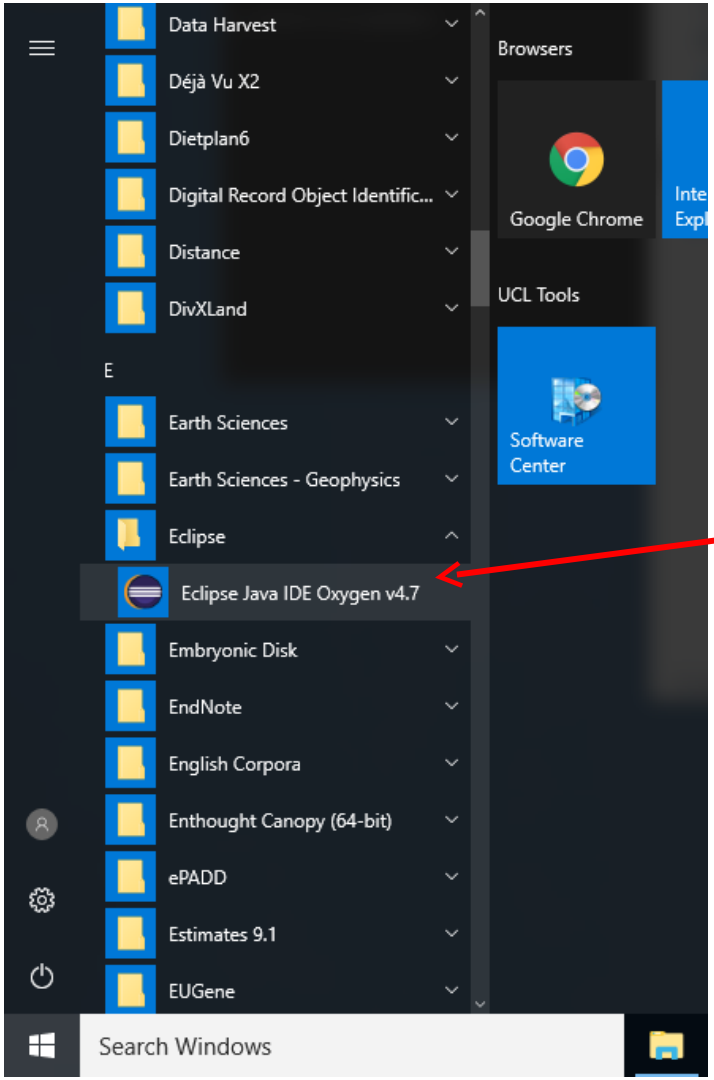
Module 1: Getting Started with Java and Eclipse

Ben Waugh & Simon Jolly

Starting Eclipse

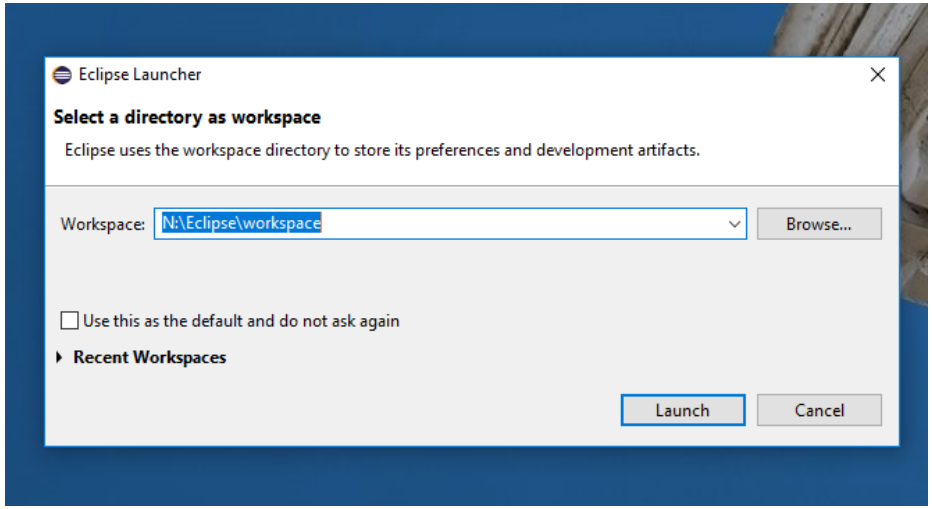
- Eclipse is the “Integrated Development Environment” (IDE) that we will use in this course.
- Other IDEs that can be used for Java include NetBeans…
- You can write Java code using a simple text editor, but an IDE gives:
  - tools to manage large numbers of classes within several projects;
  - easy access to help about Java classes and methods;
  - syntax highlighting;
  - automatic checking for errors as you type.
- Eclipse is installed on Desktop@UCL: runs locally on PC in cluster room. Avoids problems due to network congestion and memory use on shared server.
  - Desktop@UCL Anywhere has a (much) older version, not recommended.
- It is available for free download from eclipse.org for Windows, Mac OSX and Linux: look for “Eclipse IDE for Java (*not Java EE*) Developers”.

Starting Eclipse



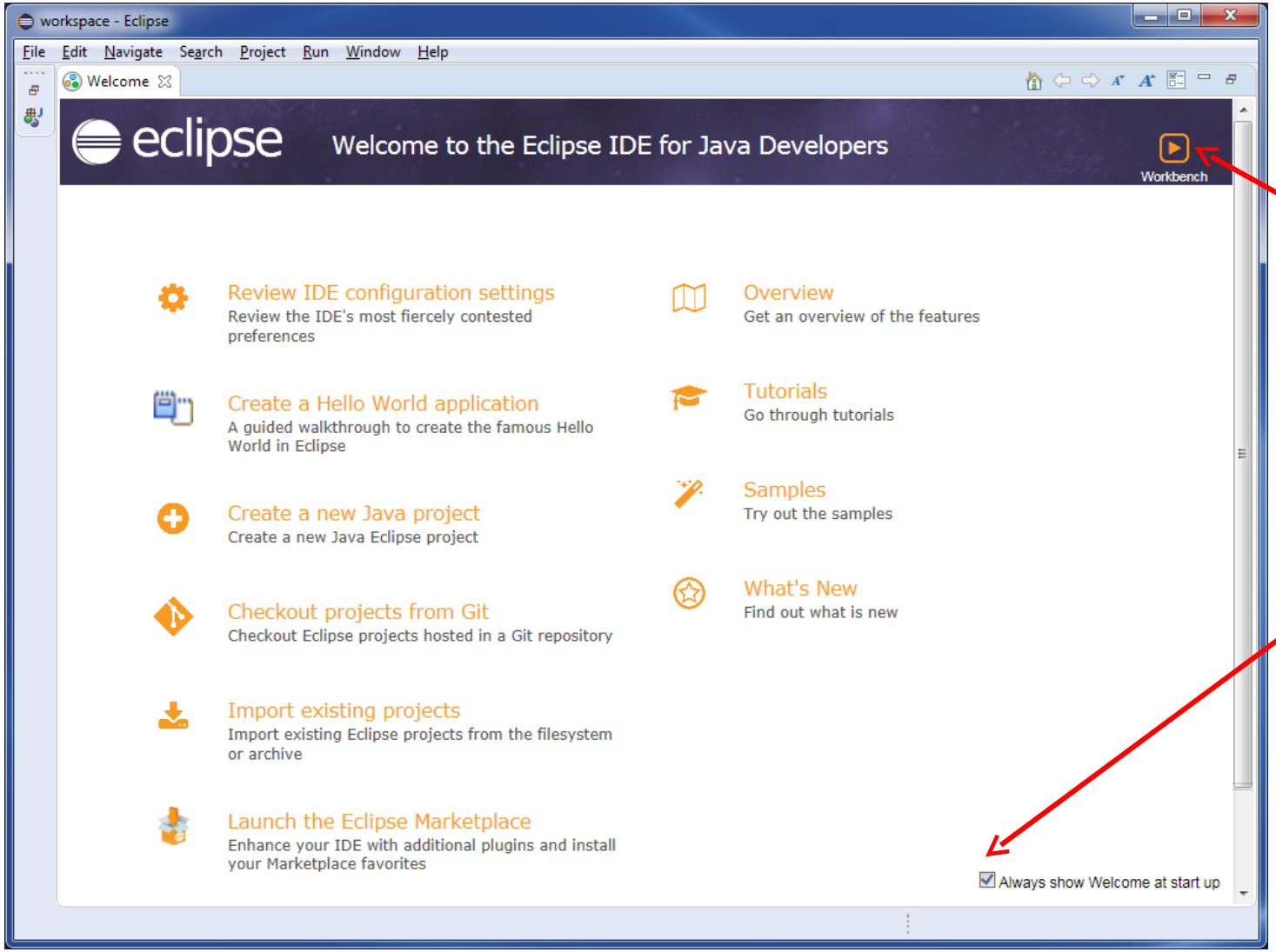
- Start typing “Eclipse” in the search box at the bottom of the screen, and select “Eclipse Java IDE Oxygen v4.7” when it appears.
- Alternatively, find “Eclipse Java IDE Oxygen v4.7” in the “Eclipse” folder in the Start menu.
- You can make it easier to start Eclipse next time by right-clicking this entry and selecting “Pin to Start”.

Starting Eclipse



- When starting Eclipse for the first time, you need to create a default workspace.
- The default workspace location can be anything on the N: drive, but the recommended location is: N:\Eclipse\workspace
- You may need to type this in by hand if the files do not show up in the file browser…

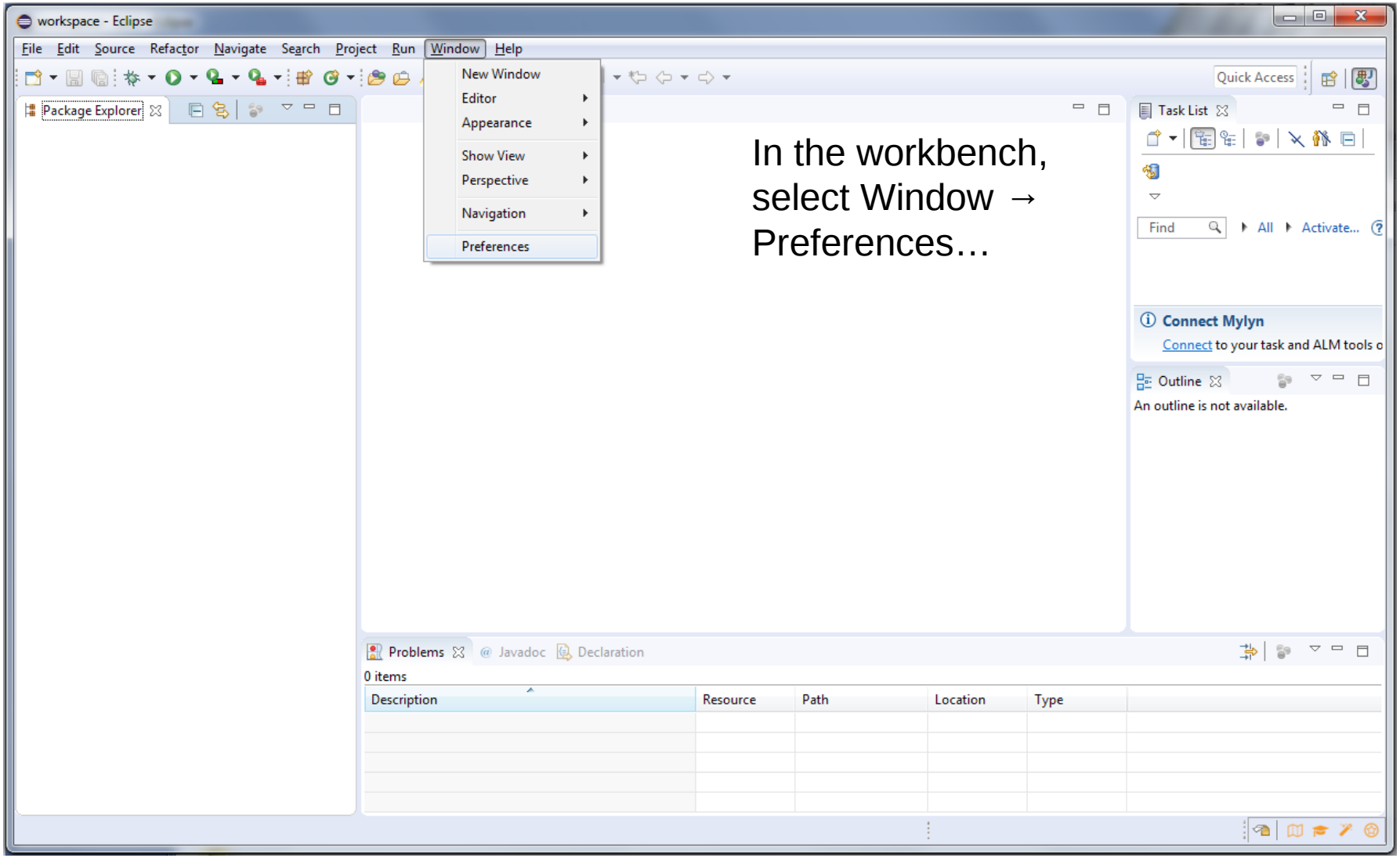
Starting Eclipse



Click here to remove the Welcome screen and go to the Workbench

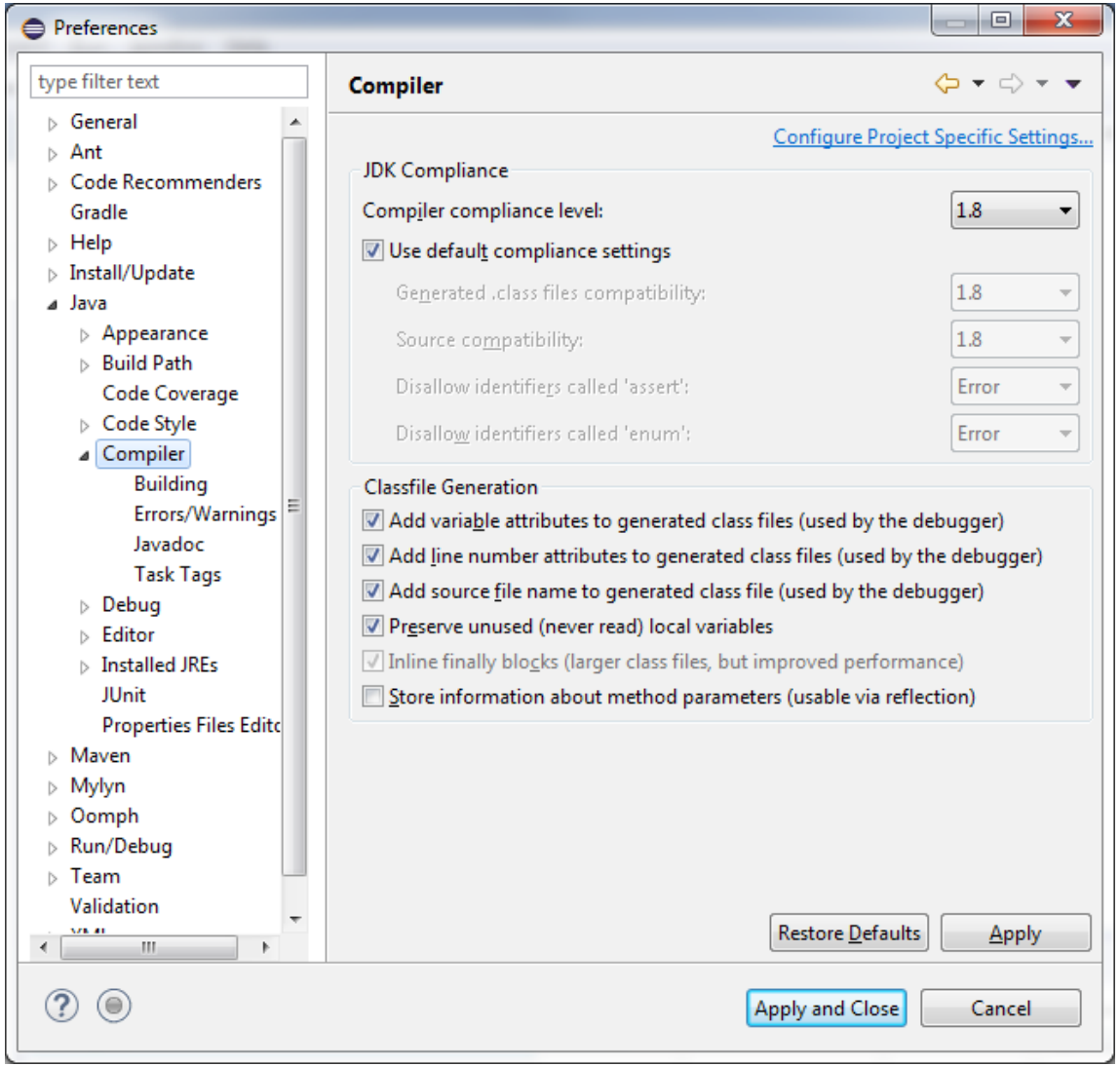
Uncheck this box if you don't want to see the Welcome screen every time on launch

Setting Up Eclipse



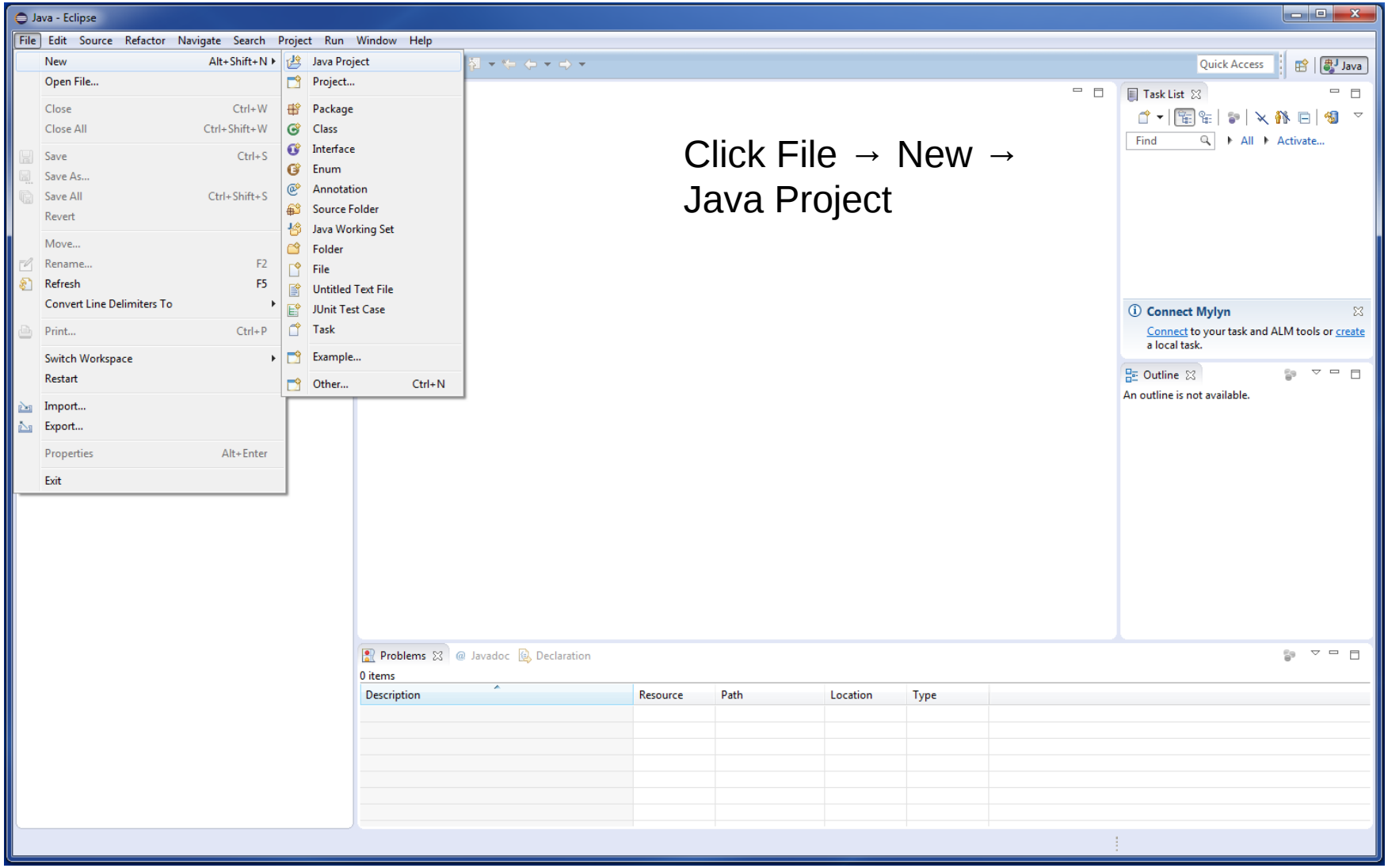
In the workbench, select Window → Preferences…

Setting Up Eclipse



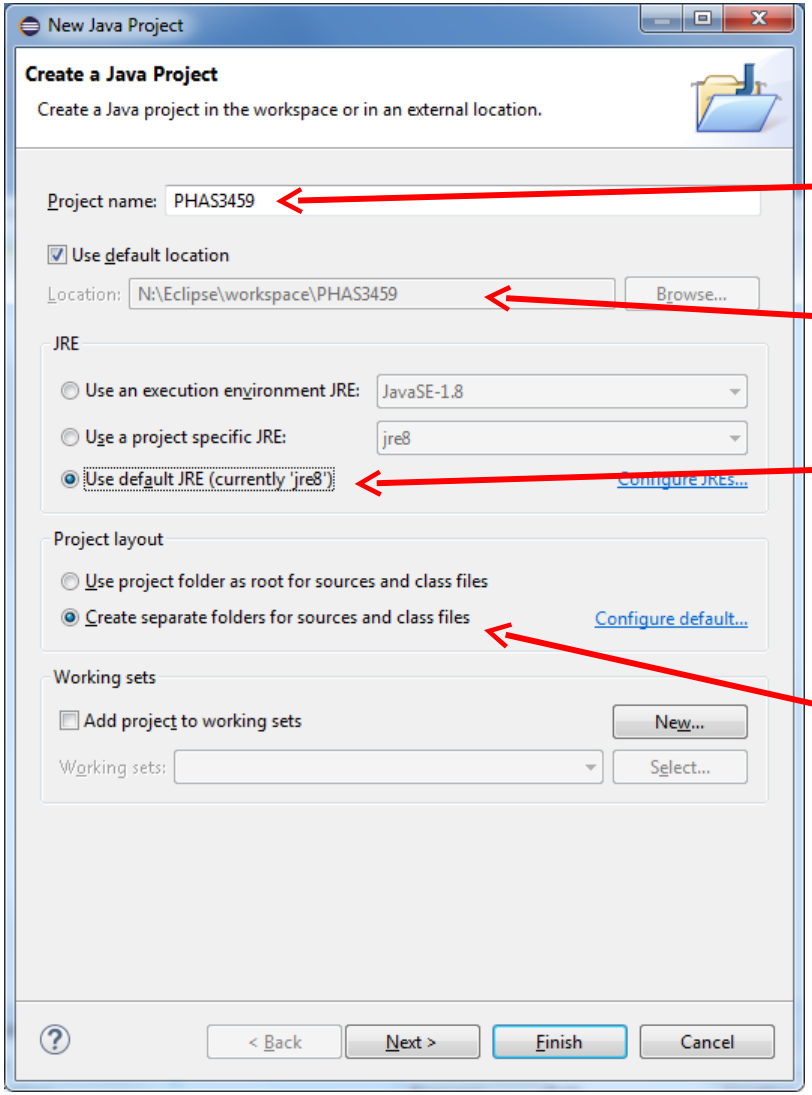
- Expand the “Java” menu item and select “Compiler” from the submenu.
- Ensure the “Compiler compliance level” is set to 1.8.
- Click “Apply and Close” or “Cancel” as appropriate.

Creating A Project



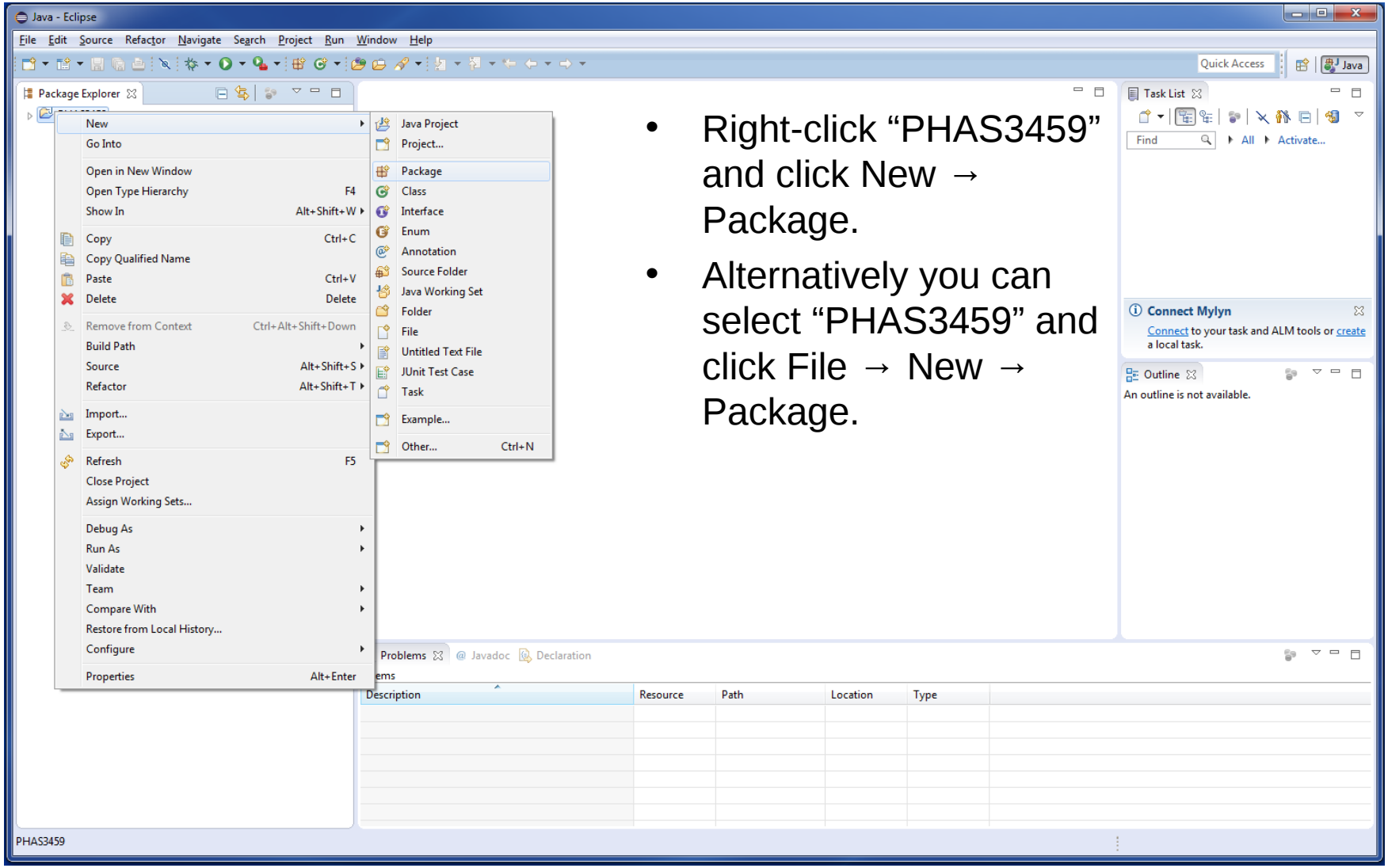
Click File → New → Java Project

Creating A Project



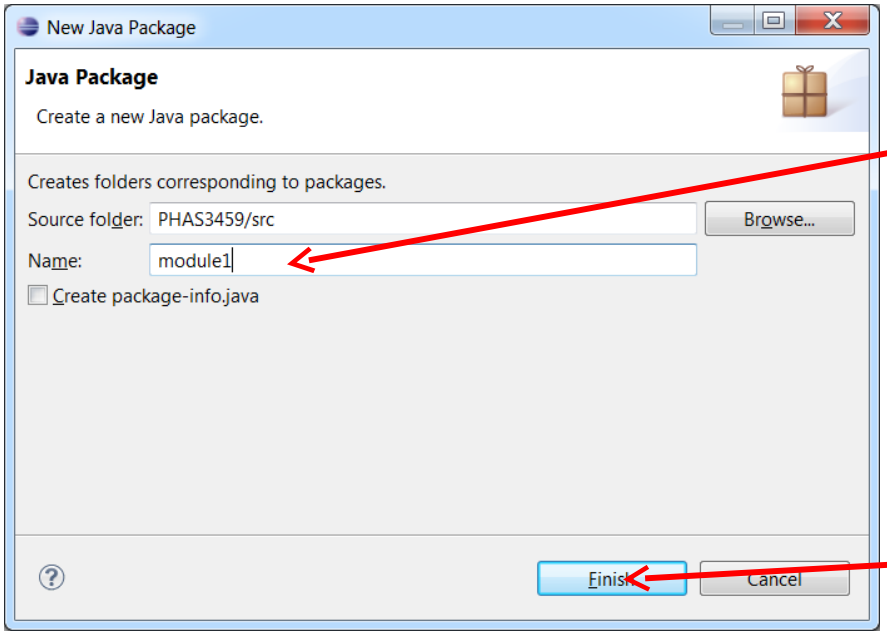
- Set project name to “PHAS3459”
- Don’t change this.
- Check JRE (Java Runtime Environment) is using default version 1.8.
- Don’t change this.
- Click “Finish” (“Next” will give you more options, which you don’t need.)

Creating Packages



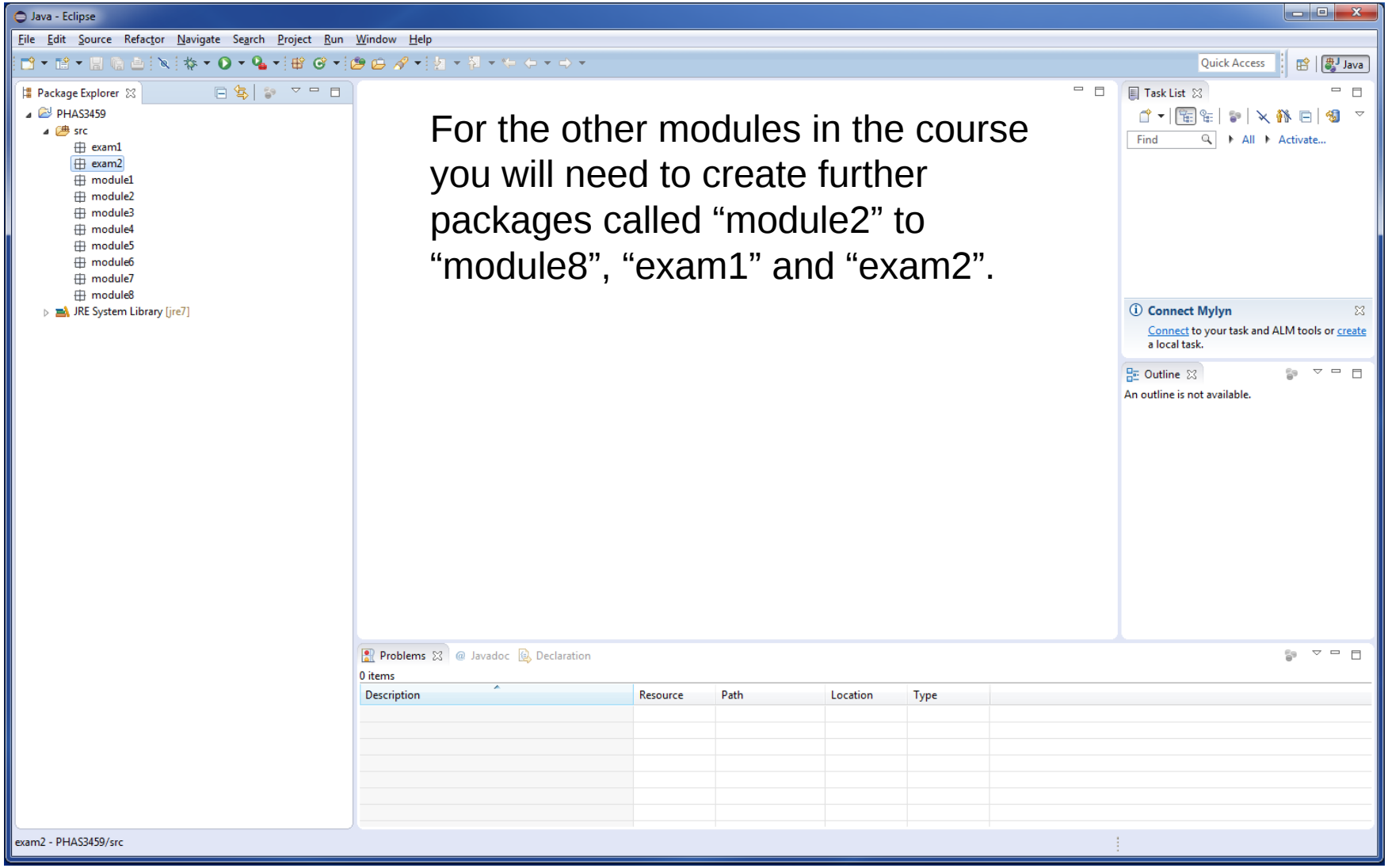
- Right-click “PHAS3459” and click New → Package.
- Alternatively you can select “PHAS3459” and click File → New → Package.

Creating Packages

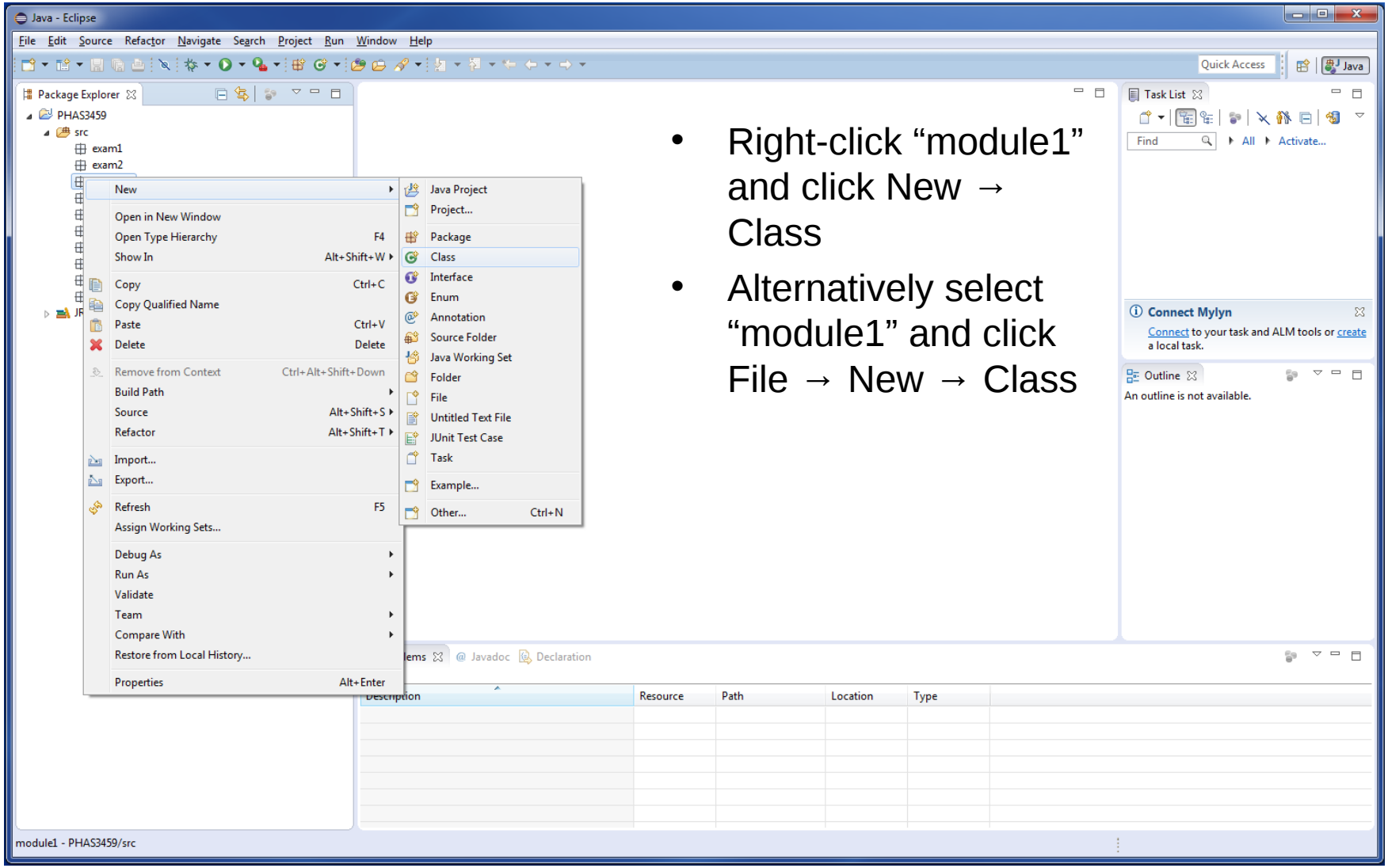


- Name your first package “module1”.
- Click “Finish”.

Creating Packages

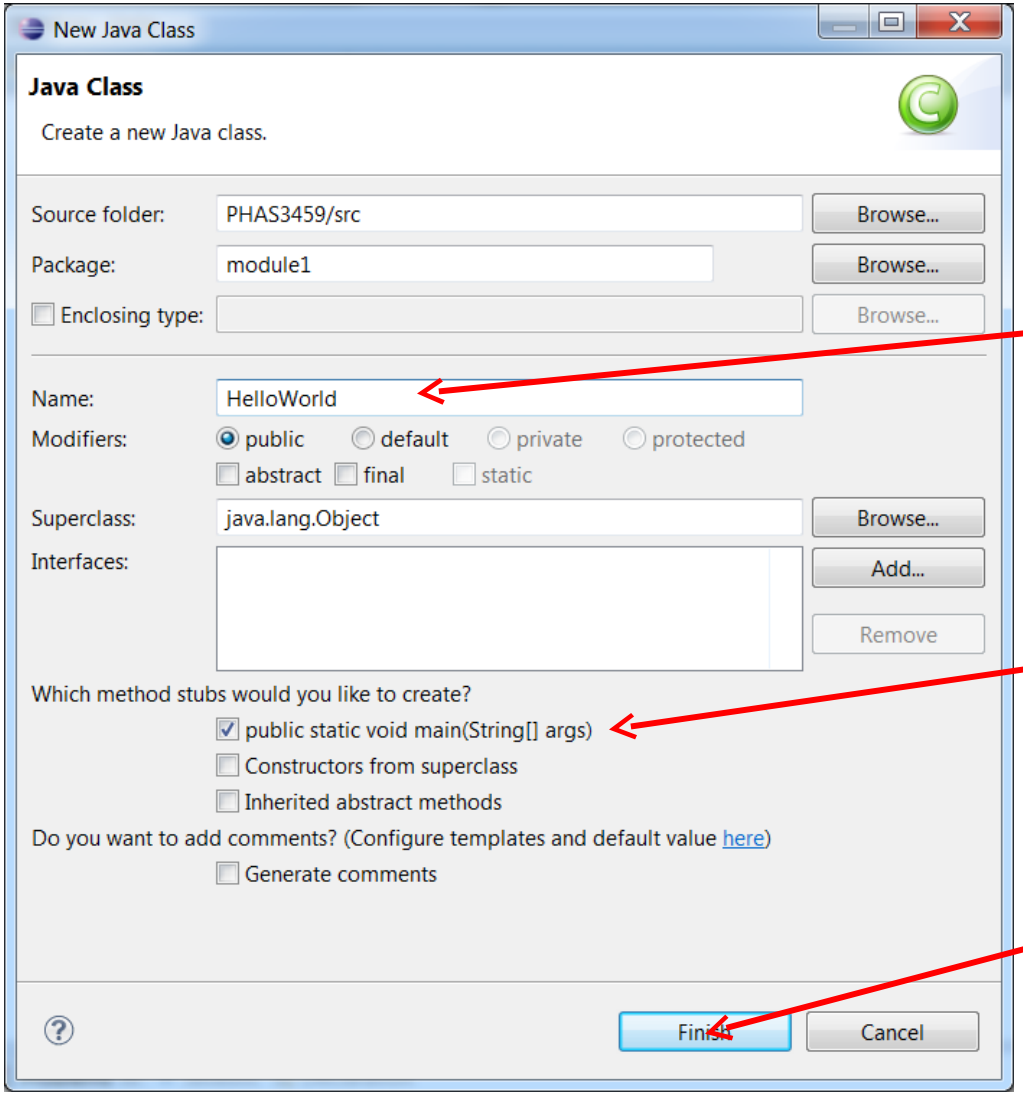


Creating A Class



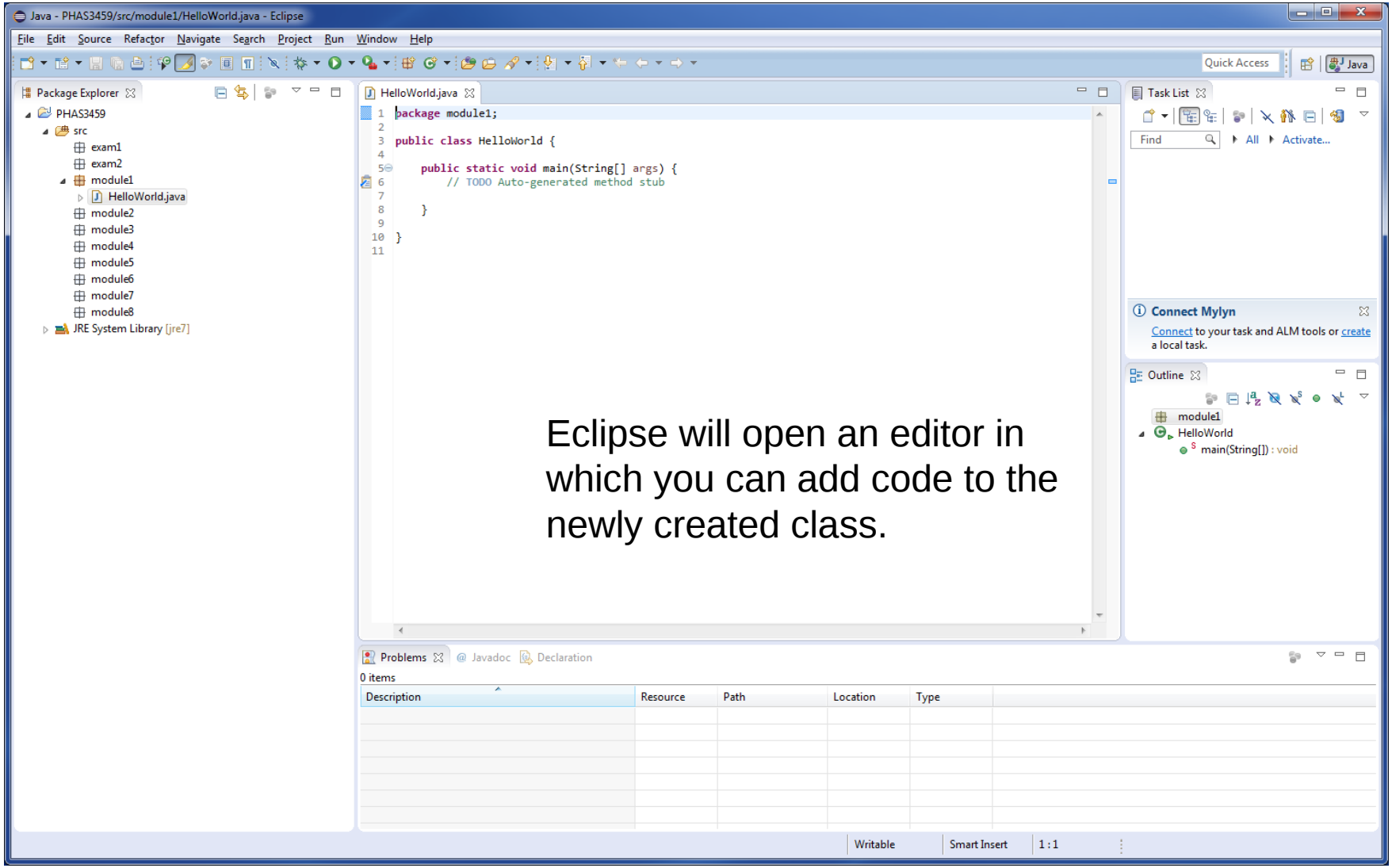
- Right-click “module1” and click New → Class
- Alternatively select “module1” and click File → New → Class

Creating A Class



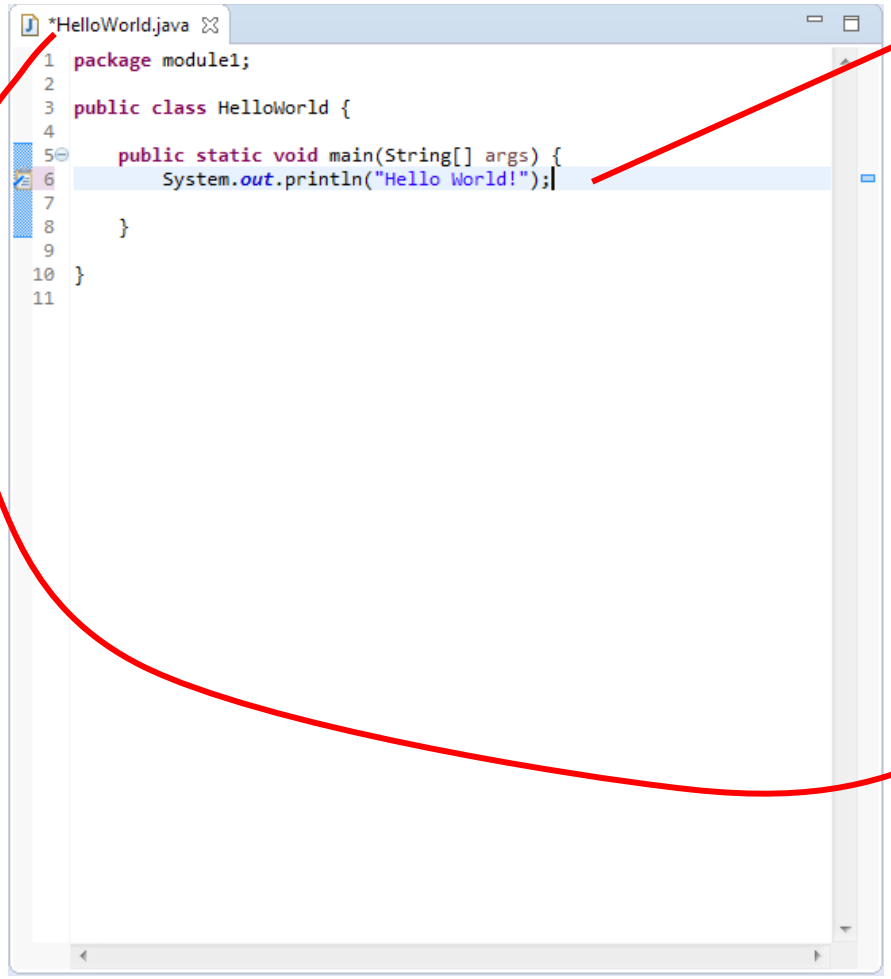
- Set Name to “HelloWorld”.
- Only the first option should be selected.
- Click “Finish”.

Creating A Class



Eclipse will open an editor in which you can add code to the newly created class.

The “hello world” Program

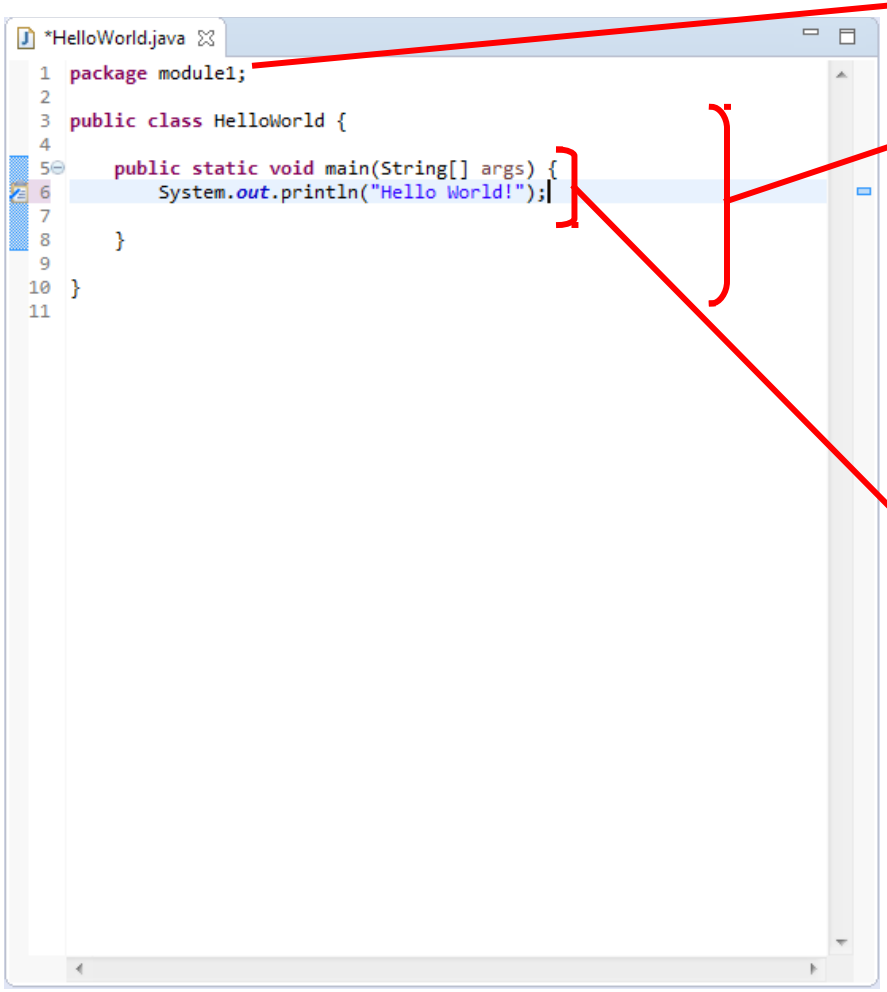


- Enter this text in the editor.
  - You will have to type it (almost) exactly as it appears here:
    - Java is case-sensitive so “`PrintLn`” is not the same as “`println`”;
    - The semicolon at the end of the line is essential.
  - As you type, Eclipse will suggest some things you might be looking for and give you access to descriptions of what they do.
- The asterisk means this file has not yet been saved. You can save it using the File menu or the keyboard shortcut ctrl-S.

Some Terminology

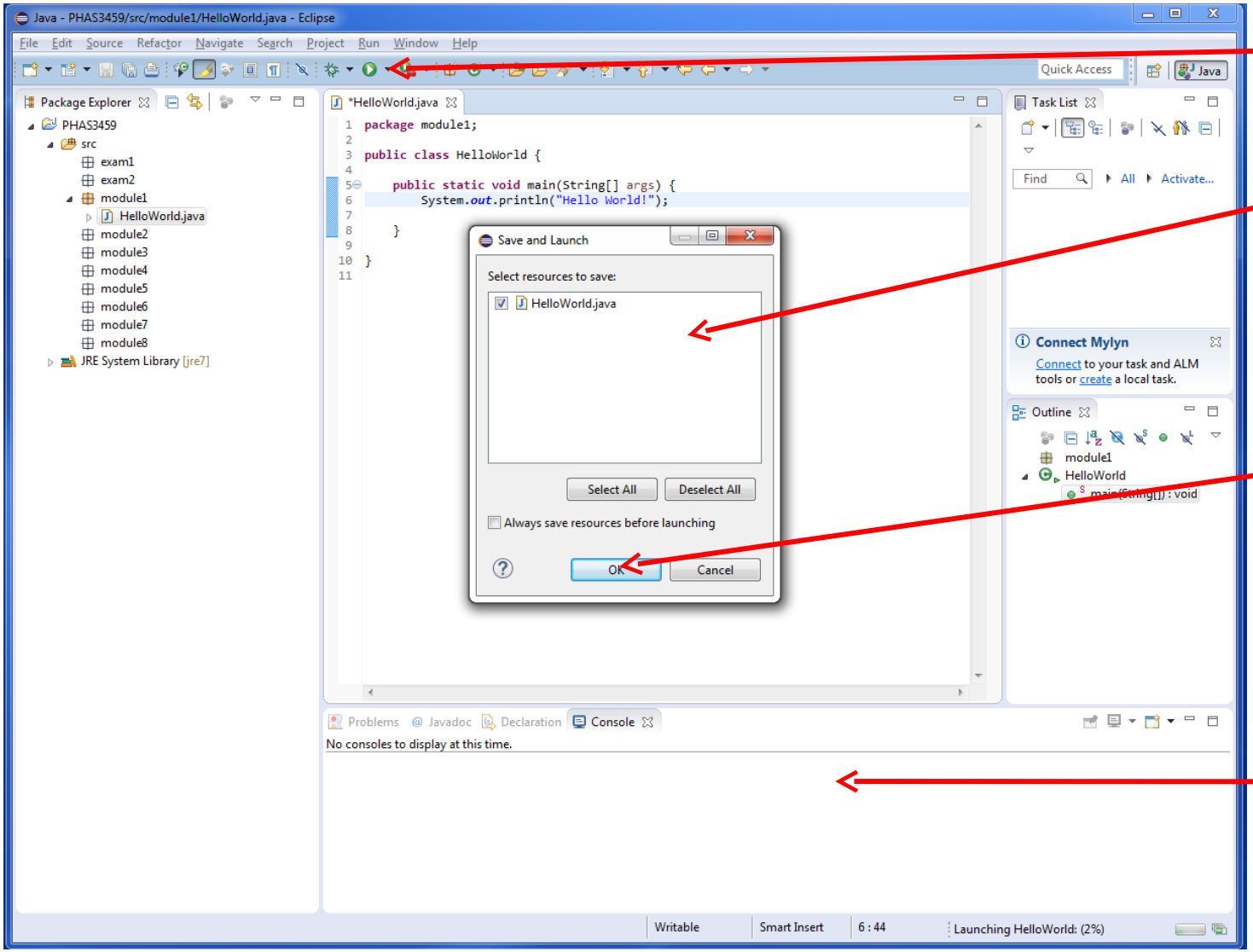
- **Class**: Every bit of Java code you write will be contained in a “class”. We will explain what this means in more detail in module 2. For now, you need to know that to run a Java program you need to create a class with a “main” method. Later you will create more complicated classes and programs that use more than one class.
- **Package**: In order to simplify the management of large and complex programs, the classes may be grouped into packages. A package contains a set of classes that are related in some way and are located within the same directory (folder) on the computer.
- **Project**: In Eclipse a “project” may contain several packages that are part of a larger program or set of programs.
- In this course you will use a single project, with a separate package for each module. Within each of these modules you will in general create several classes.

The “hello world” Program



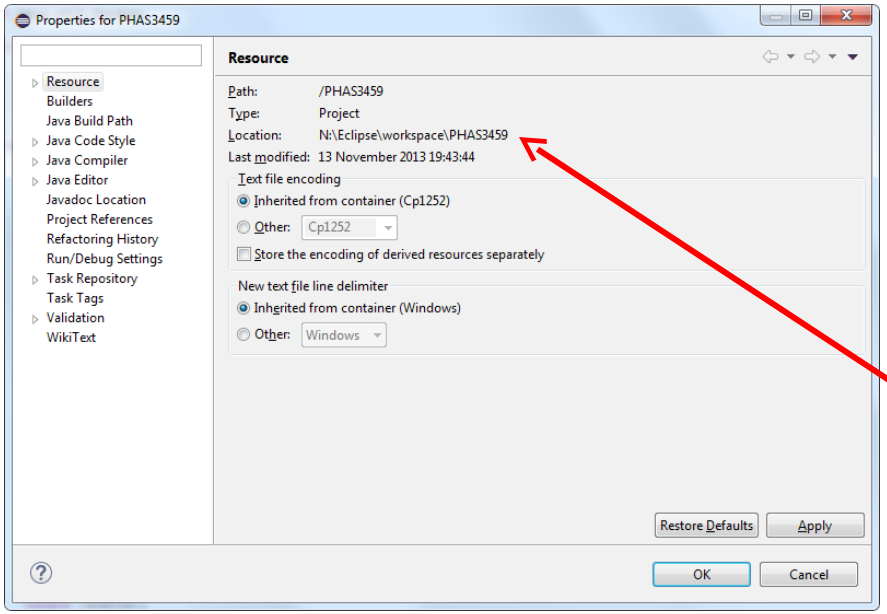
- This class is in the package called “module1”.
- Everything here is part of the HelloWorld class.
- **Method**: A class may contain several methods or functions. In Java these words mean the same thing, but some languages make a distinction.
- The HelloWorld class has only one method, the “main” method.
- This is a special method: it tells the computer what to do when you run your Java program.
- We’ll add more methods later and see how they can call each other.

Running the “hello world” Program



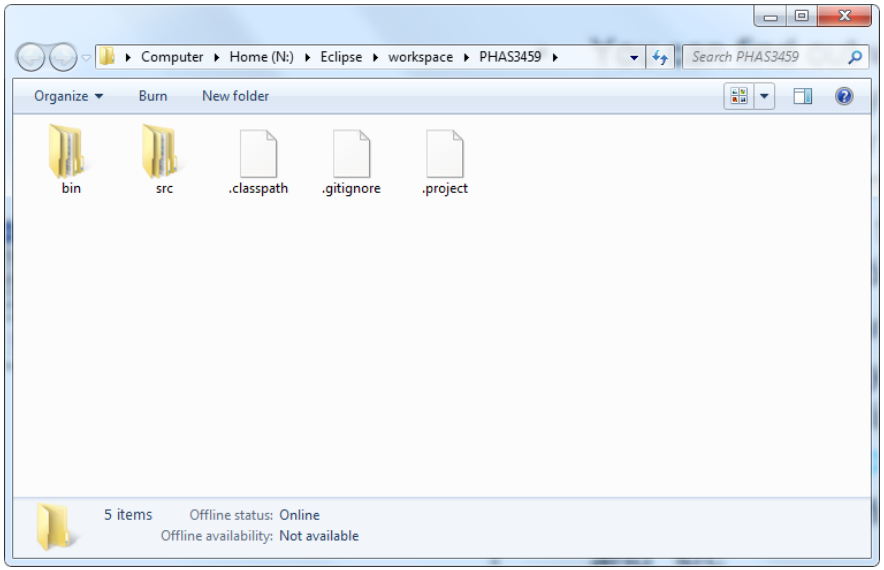
- Click the white/green “Run” arrow.
- If you haven’t already saved the file, you will get this dialogue box:
  - Tick the box and click “OK”.
- The output of your program appears here.

Java Files



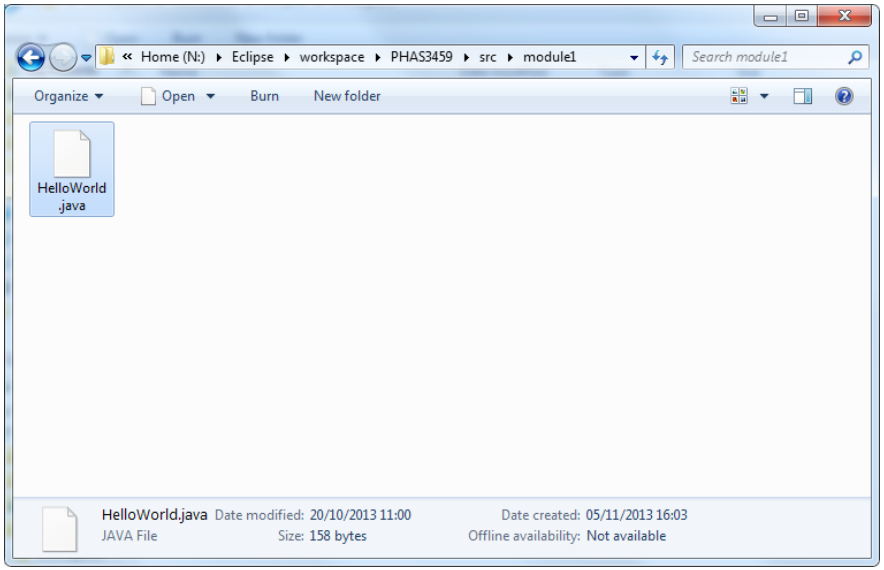
- You can find out where your Java files are stored by right-clicking on the project or any of the packages or classes in it, and selecting “Properties” from the menu
- Under the “Resource” tab, the location of the source files is shown next to “Location”.

Java Files



- They should be within the directory you specified as your workspace when starting Eclipse, e.g. in N:\Eclipse\workspace\PHAS3459.
- You will find two folders called “bin” and “src”.

Java Files



- In the “src” folder you will find a folder for each package in your project
- In these you will find “.java” files, which contain the source code (i.e. what you write) and are the files that **should be uploaded** when submitting work.
- In the “bin” folder you will find “.class” files, which are the compiled “byte code” files that are run by the Java Virtual Machine. **Don’t upload them! If you only upload “.class” files, you won’t get any marks as we can’t see your code!**
- Make a shortcut on the desktop to the folder with your Java files!

Default Workspace

- When you launch Eclipse, the Workspace Launcher will give you the option to set the current choice as default.
- If you check this box and your N: drive is not available on launch, you will get an error.
- You can revert to the old behaviour by selecting “Prompt for workspace on startup” under “Preferences → General → Startup and Shutdown → Workspaces”.