Getting Started

(http://docs.oracle.com/javase/tutorial/getStarted/index.html)

Objectives

- About the Java Technology
- What can Java Technology do?
- How can Java support platform-independence?
- Java Platform
- Set up Environment Variables
- The first Java program in the NetBeans
- Structure of a Java program.
- End users run Java Programs

About the Java Technology(1)

- ***History***
  - 1990, James Gosling, Bill Joy, Patrick Naughton(Sun Microsystem) developed the Oak language for embedding programs to devices such as VCR, PDA (personal data assistant). The Oak programs require:

    - Platform independent/- Extremely reliable/ - Compact
  - 1993, interactive TV and PDA failed, Internet and Web were introduced, **Sun** change the Oak to an internet-development environment with a new project, named **Java**.
  - 1994, the Sun's *HotJava Browser was introduced* (written using Java). It showed the strength of Java applets and abilities to develop Java application.

About the Java Technology(2)

History…

- **Embedded Systems (1991 – 1994)**
- **A client – side Wonder (1995 – 1997)**
- **Moved into the Middle – tier (1997 – to present)**
- **Future: may gain more success**

About the Java Technology(3)

The Java Programming Language is a high-level language. It's **characteristics**:

- Simple
- Object oriented
- Distributed
- Multithreaded
- Dynamic linking
- Architecture neutral
- Portable
- High performance
- Robust
- Secure

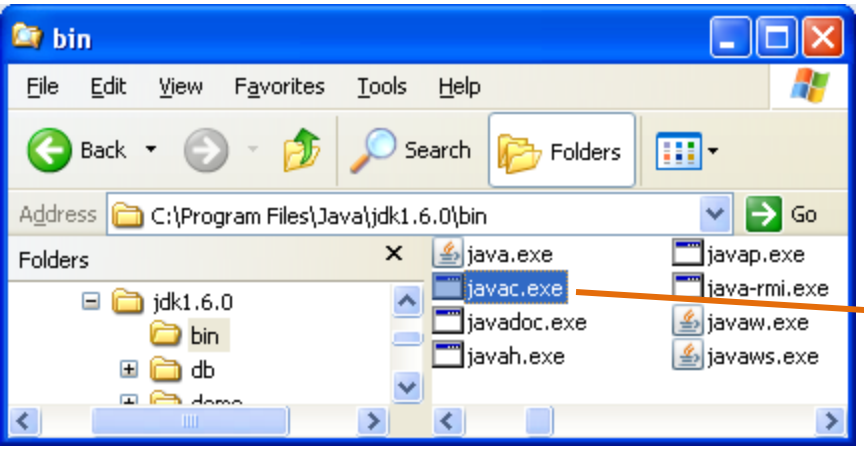
What can Java Technology do?

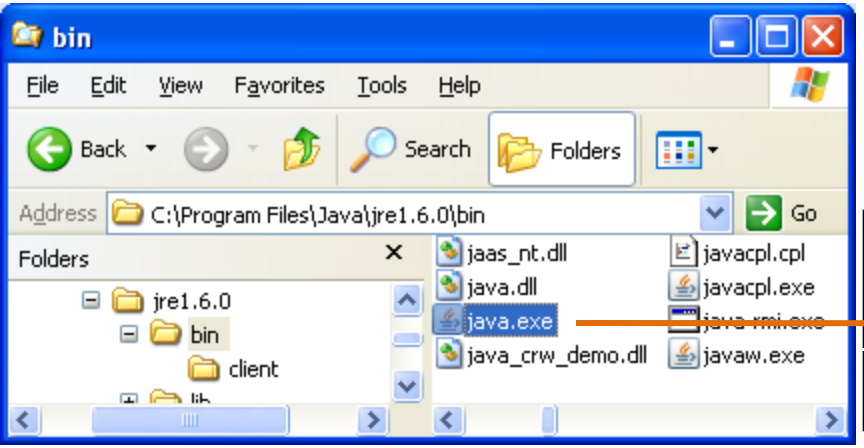
Using Java, we can:

- Development Tools.
- Application Programming Interface (API).
- Deployment Technologies.
- User Interface Toolkits.
- Integration Libraries.

➔Desktop Application (Console App, GUI Apps)
➔Web-based Applications
➔ Network-based Applications
➔ Game
➔ Distributed Applications
➔ Embedding Application (Apps on Devices)

How can Java support platform-independence?





file.java (plain text)

↓

Java Compiler
Javac.exe

↓

Platform-Independent

Java byte-code: **file.class**

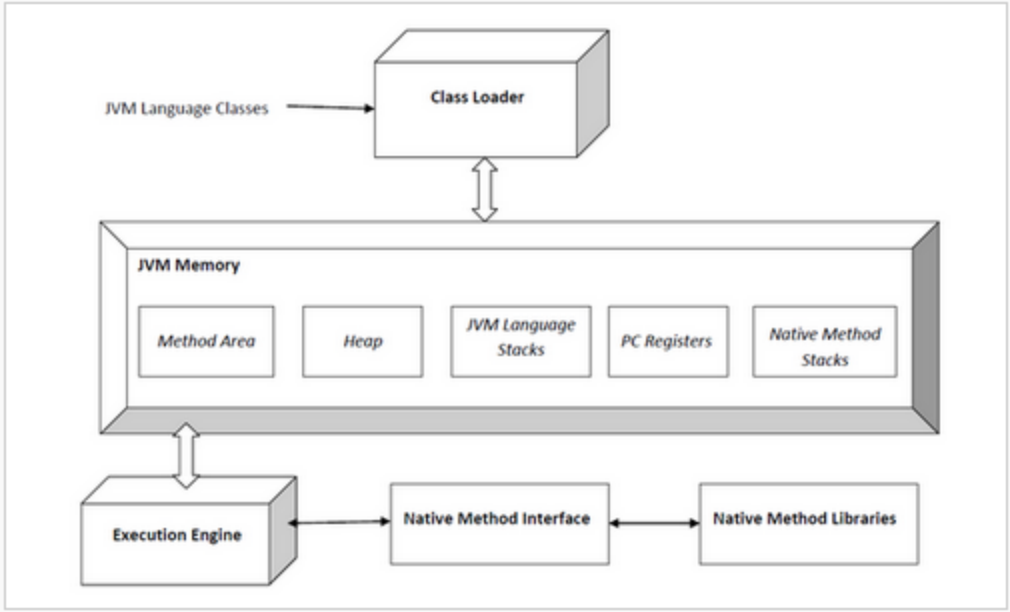
↓

Java Runtime Interpreter / Java Virtual Machine (java.exe)

IBM | **Macintosh** | **Sparc**

Java Virtual Machine

- The Java Virtual Machine is an abstract computing machine. Like a real computing machine, it has an instruction set and manipulates various memory areas at run time. It is reasonably common to implement a programming language using a virtual machine; the best-known virtual machine may be the P-Code machine of UCSD Pascal.



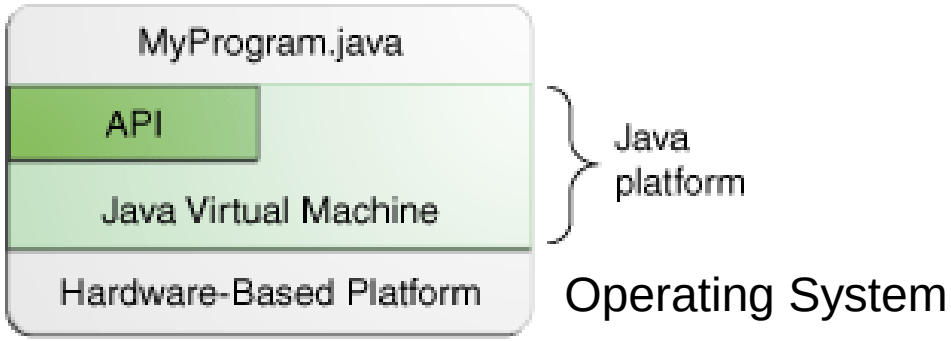
Overview of a Java virtual machine (JVM) architecture based on The Java Virtual Machine Specification Java SE 7 Edition

http://en.wikipedia.org/wiki/Java_virtual_machine

More details:
https://docs.oracle.com/javase/specs/jvms/se8/html/jvms-1.html#jvms-1.1

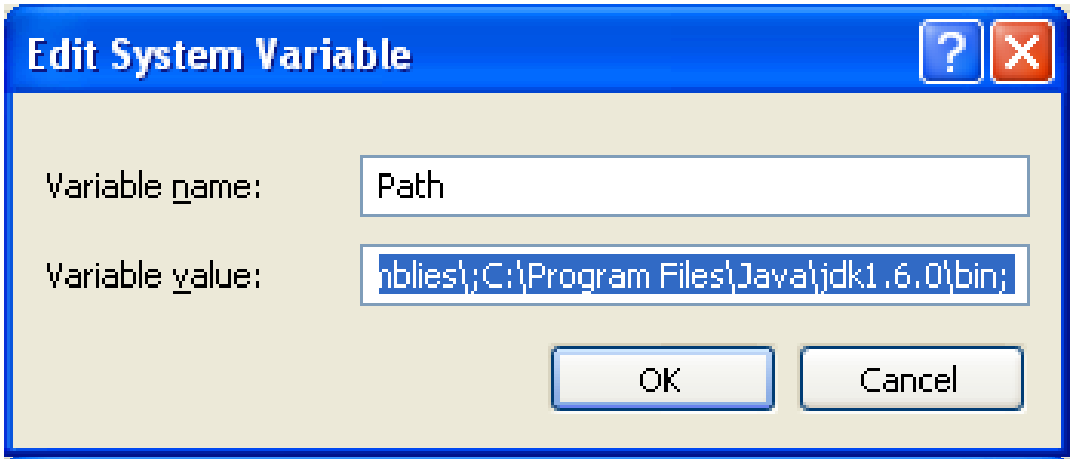
Java Platform

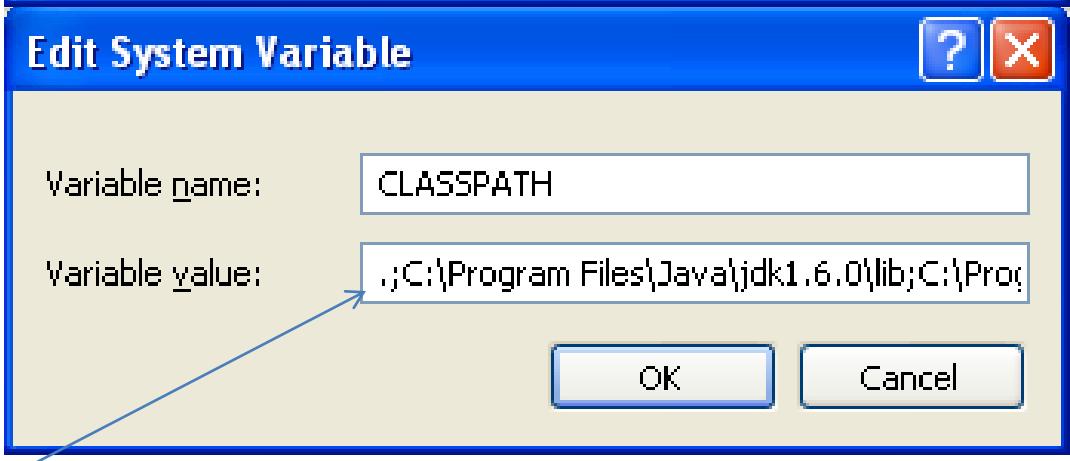
- A platform is the hardware or software environment in which a program runs.
- The Java platform has two components:
  - The Java Virtual Machine
  - The Java **A**pplication **P**rogramming **I**nterface (API)



Set up Environment Variables

- After installing JavaSE (Java Development Kit Standard Edition), environment variables should be setup to point to the folder in which JavaSE is installed.
- Steps: My Computer/ Properties/ Advanced/Environment Variables/System Variables/ Path/ Edit
- **Why?**





The point at the beginning of the CLASSPATH means that classes will be searched first in the current working folder.

The first Java program in the NetBeans

This program will show the string “Hello World” to the screen.

<u>Steps</u>

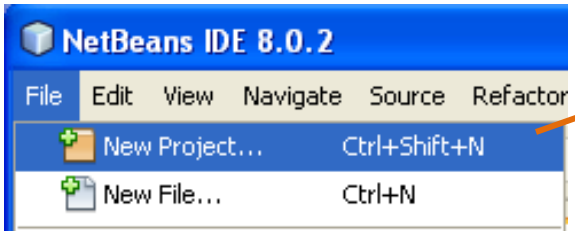
1- Create a new Java NetBeans project

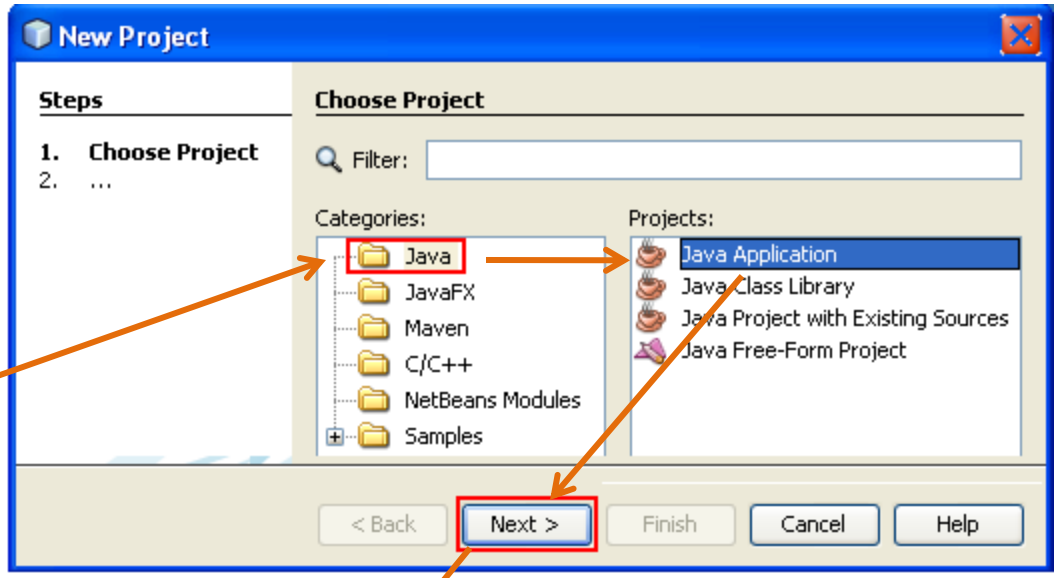
2- Add a Java class

3- Write code

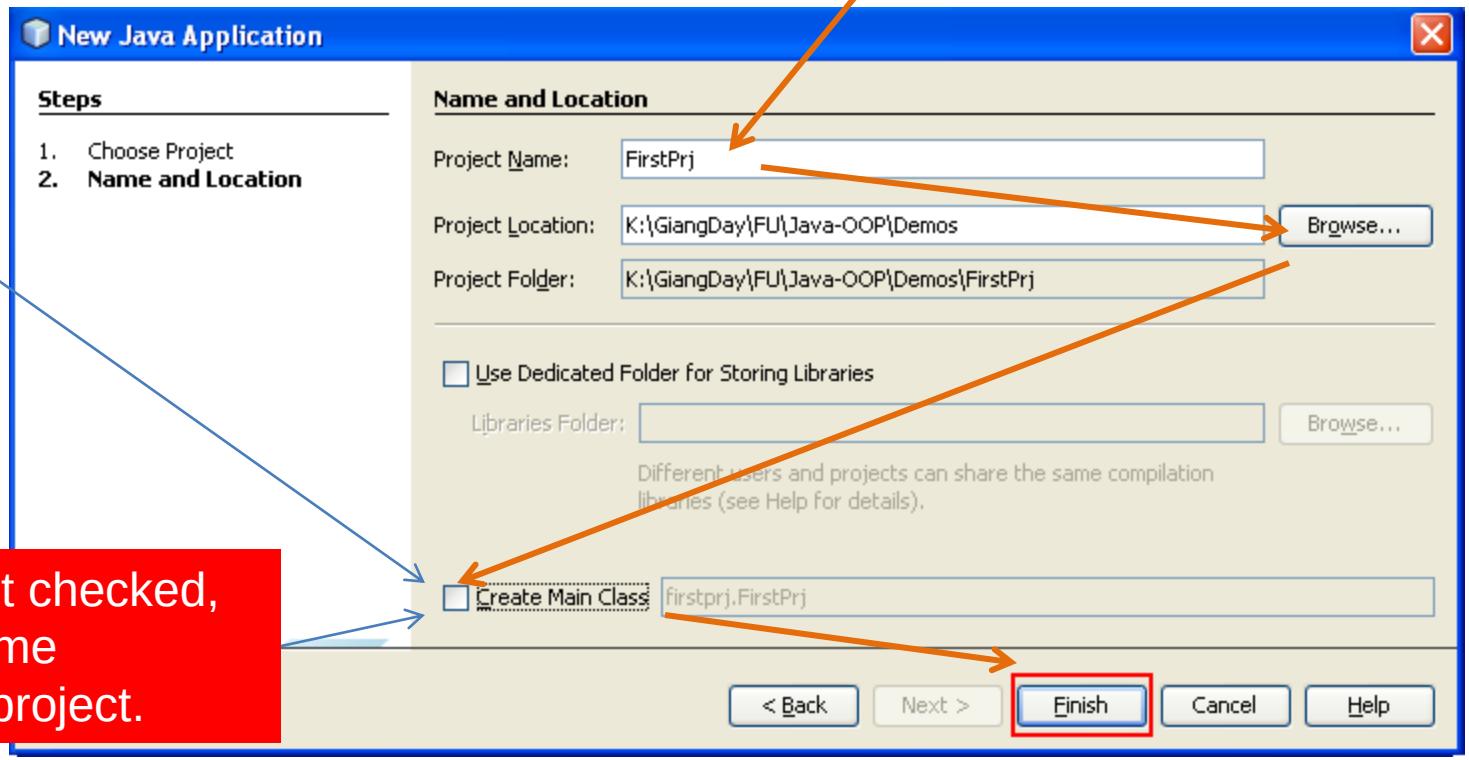
4- Compile/Run the program

Step 1- New Project



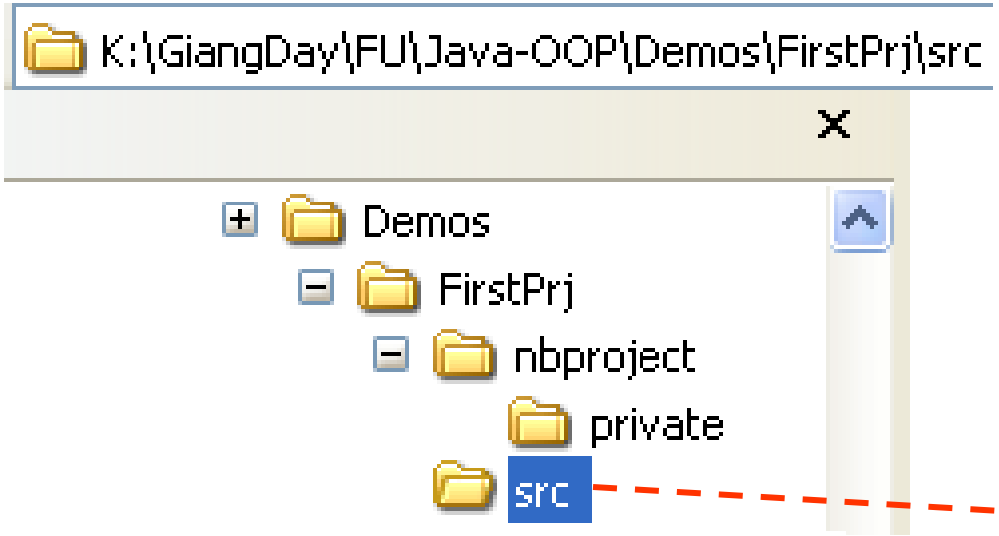


If this option is checked, NetBeans will automatically generate a class, named Main, for the project.

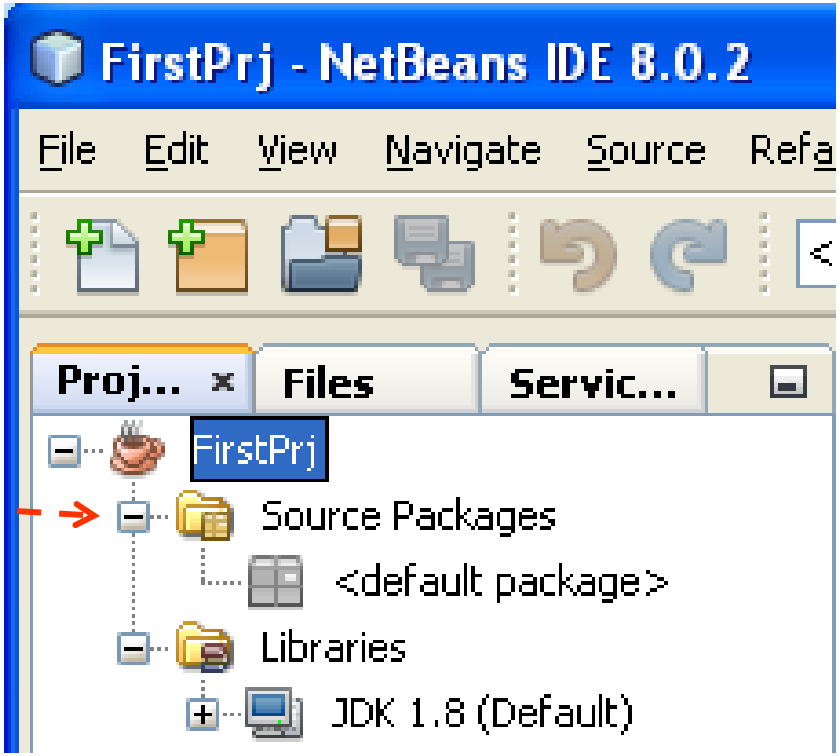


If this option is not checked, we can create some programs in one project.

New Project…: Initial Project Structure

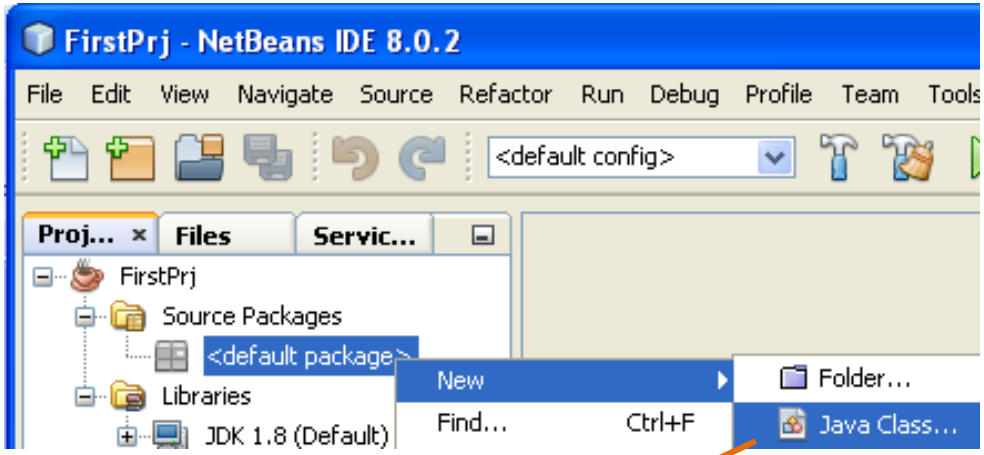


In Windows Explorer

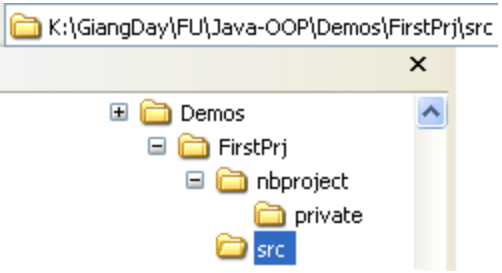


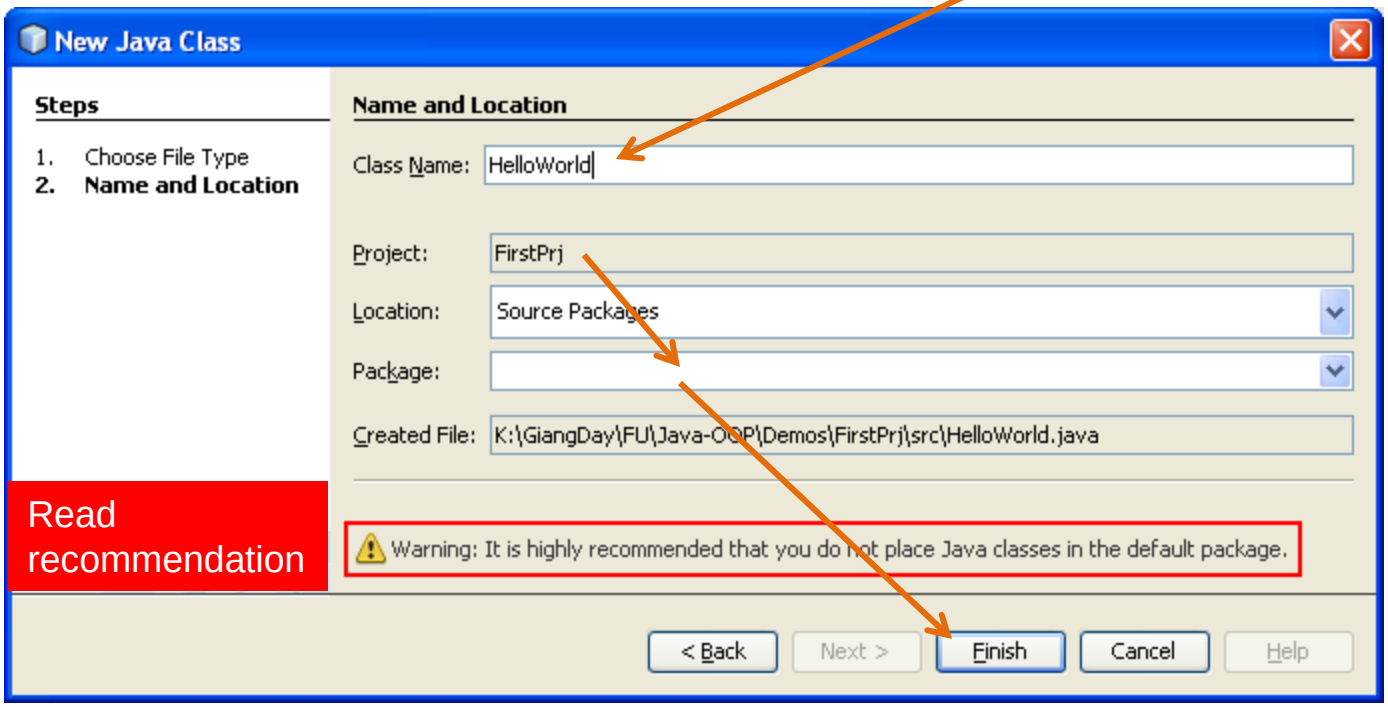
In NetBeans

Step 2: Add a Java Class



Package: Subdirectory of the folder Project/**SRC**

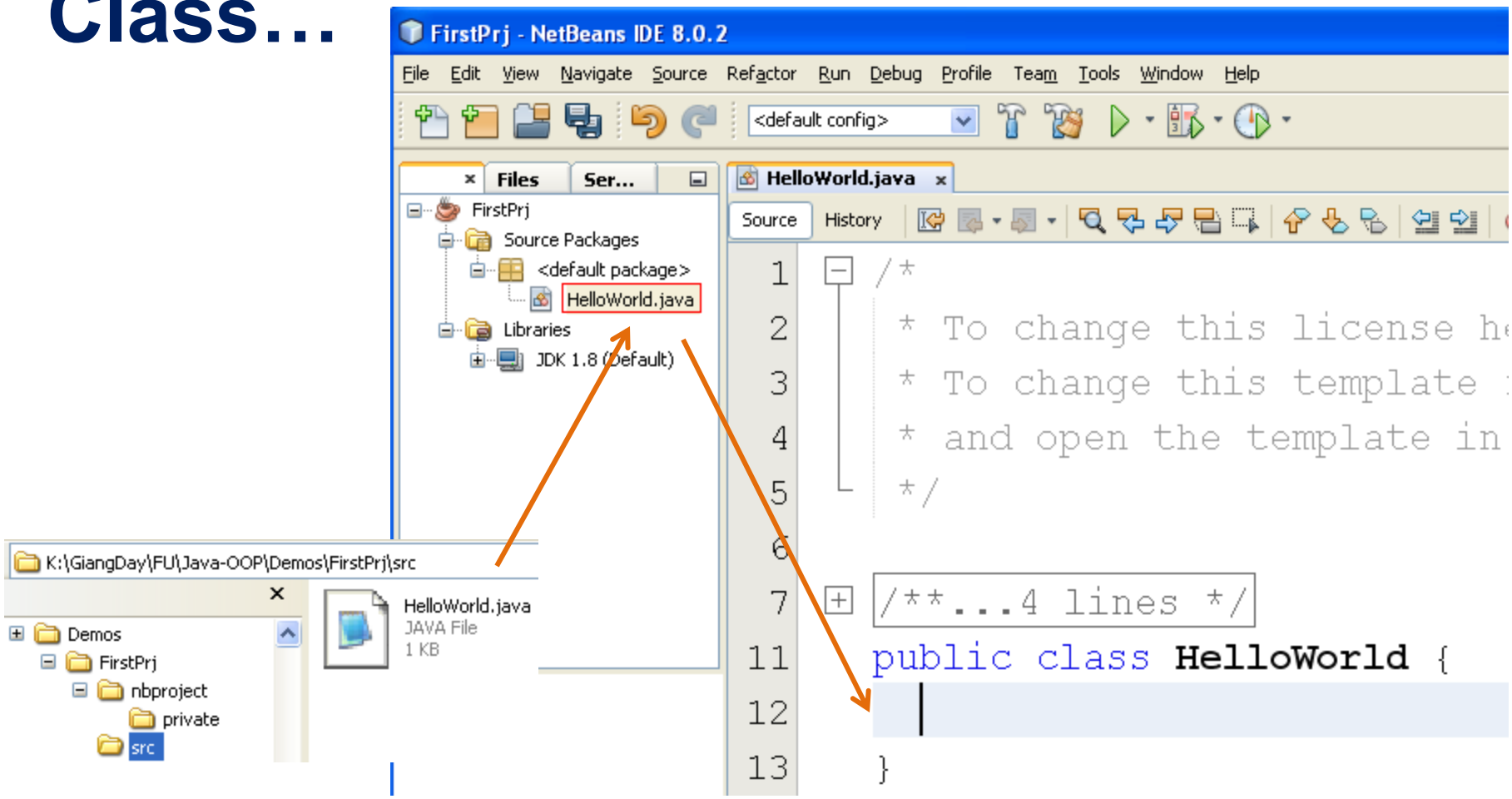




Read recommendation

In this demo, we do not specify package intentionally

Add a Java Class…



In Windows Explorer

In NetBeans

Step 3: Write code

FirstPrj - NetBeans IDE 8.0.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

<default config>

Files | Ser...

- FirstPrj
  - Source Packages
    - <default package>
      - HelloWorld.java
  - Libraries
    - JDK 1.8 (Default)

Navigator

Members

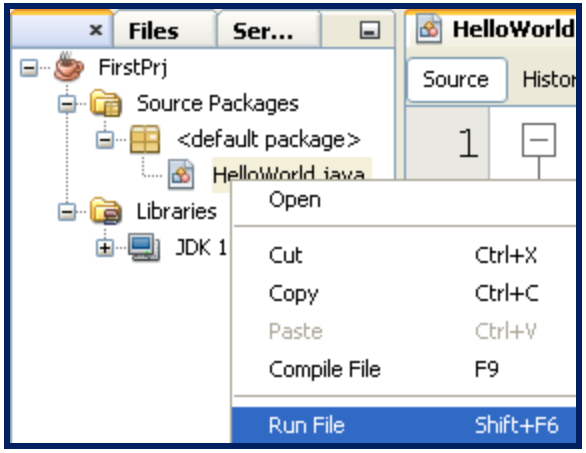
- HelloWorld
  - main(String[] args)

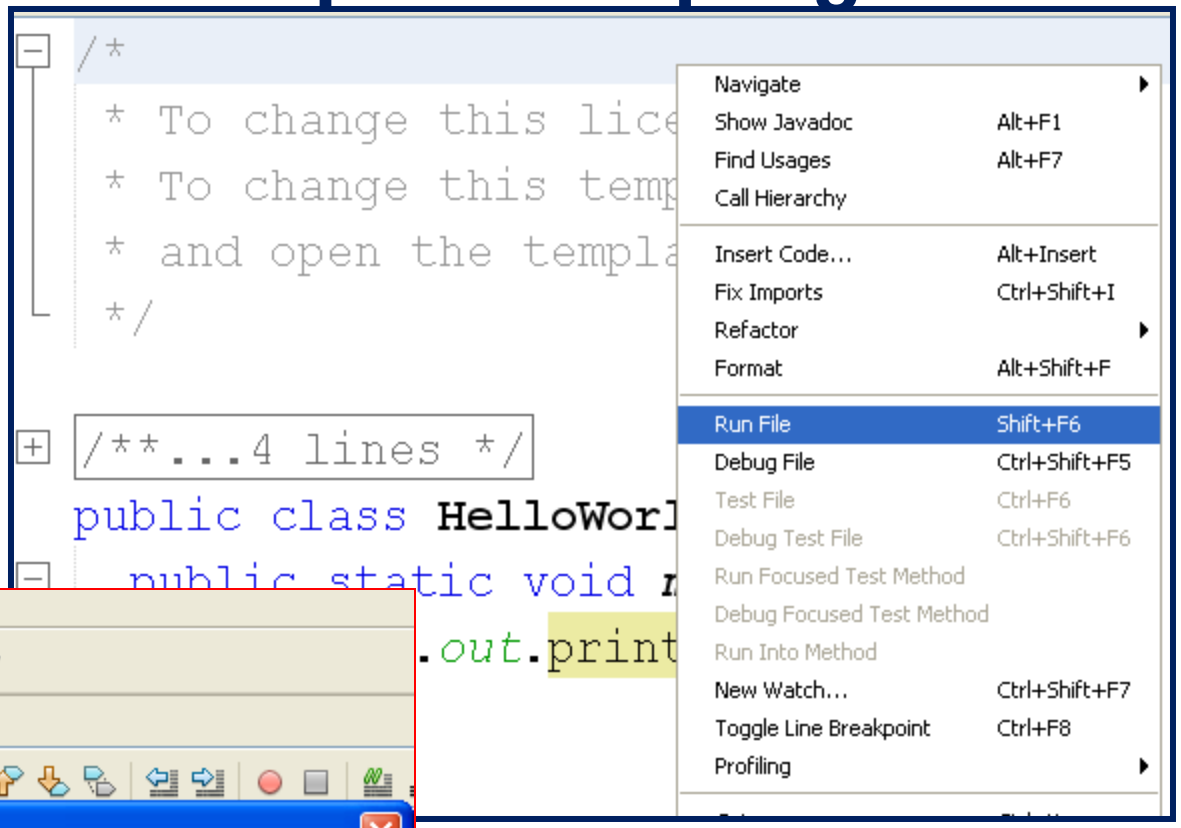
HelloWorld.java

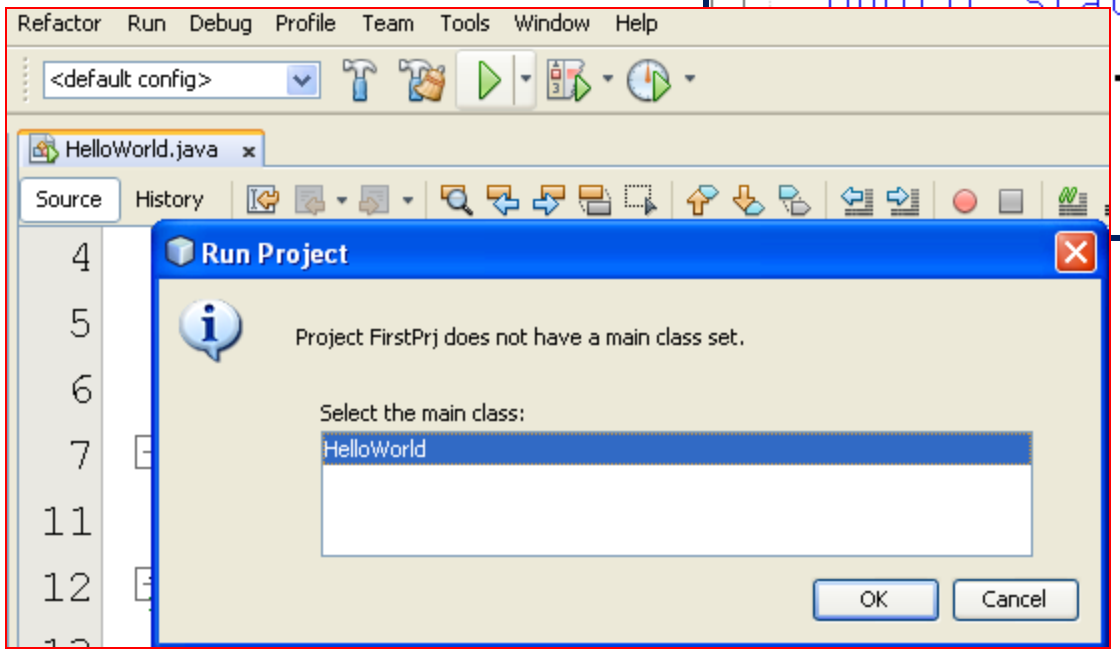
Source | History

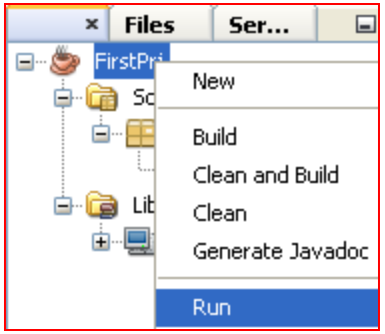
```
1  /*
2   * To change this license header, choose I
3   * To change this template file, choose To
4   * and open the template in the editor.
5   */
6
7  /**...4 lines */
11 public class HelloWorld {
12     public static void main(String[] args){
13         System.out.println("Hello World!");
14     }
15 }
```

Step 4: 4 ways to Compile/Run program in NetBeans

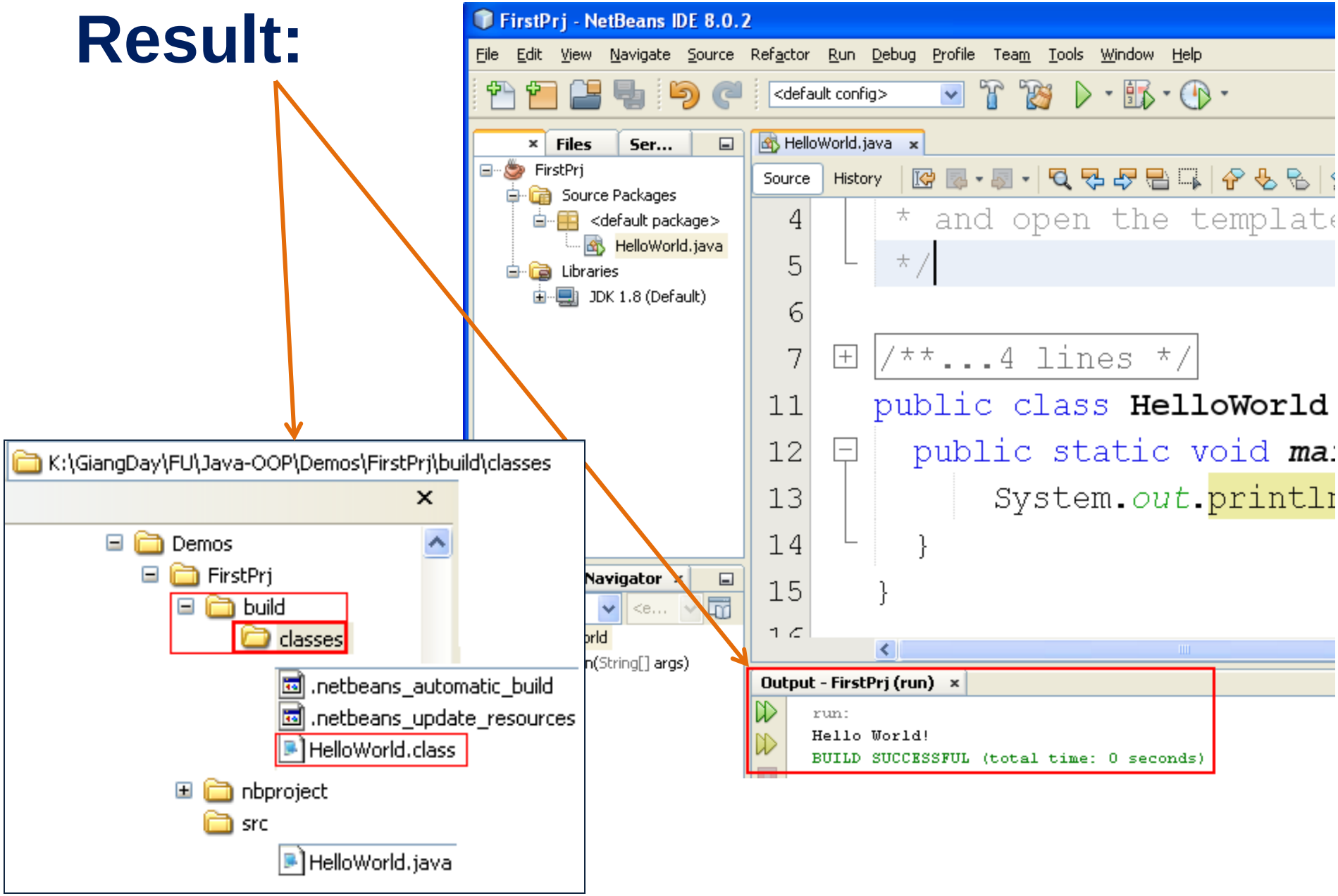








Result:


FirstPrj - NetBeans IDE 8.0.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Files
Ser...
FirstPrj
Source Packages
<default package>
HelloWorld.java
Libraries
JDK 1.8 (Default)
HelloWorld.java
Source
History
4 * and open the template
5 */
6
7 /**...4 lines */
11 public class HelloWorld
12 public static void ma
13 System.out.printl
14 }
15 }
Navigator
Output - FirstPrj (run)
run:
Hello World!
BUILD SUCCESSFUL (total time: 0 seconds)
K:\GiangDay\FU\Java-OOP\Demos\FirstPrj\build\classes
Demos
FirstPrj
build
classes
.netbeans_automatic_build
.netbeans_update_resources
HelloWorld.class
nbproject
src
HelloWorld.java

End users run Java Programs

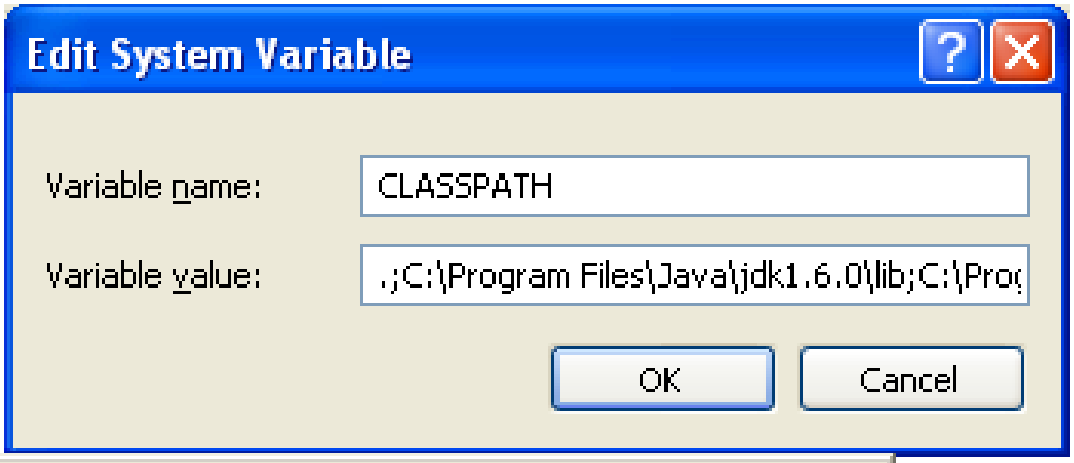
- Users can not run Java programs in NetBeans but in Java Runtime Environment (jre) installed (Java.exe) and related files
- Syntax for running a Java program:

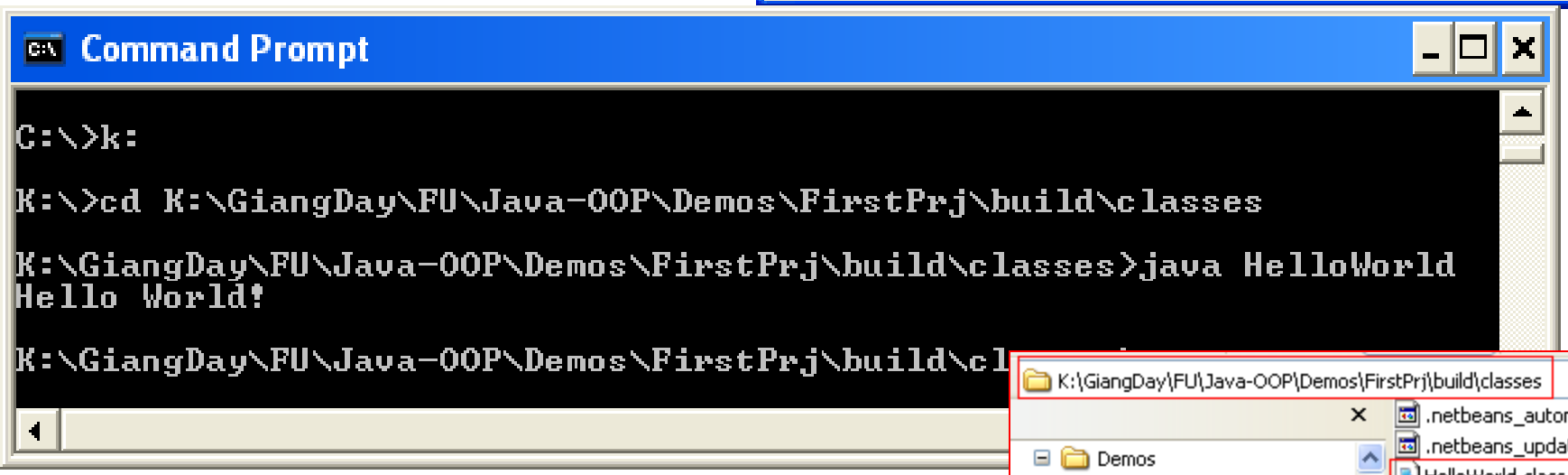
```
Command Prompt
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\USER>cd\

C:\>java -cp K:\GiangDay\FU\Java-OOP\Demos\FirstPrj\build\classes HelloWorld
Hello World!

C:\>java -classpath K:\GiangDay\FU\Java-OOP\Demos\FirstPrj\build\classes HelloWorld
Hello World!
```

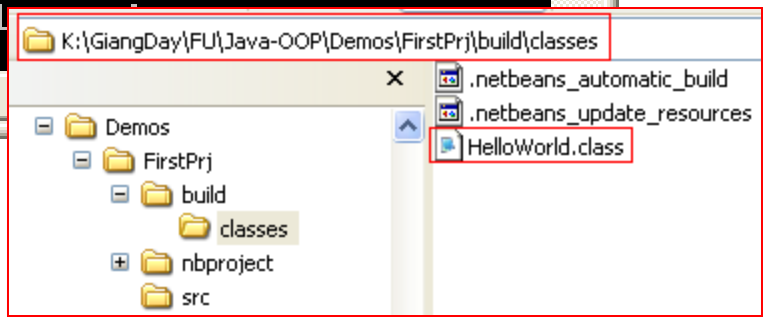
Re-try it using **Helloworld, helloworld** → Give comments

End users run Java Programs…

- If the environment variable was setup with ".;", we can run it at the working folder as:



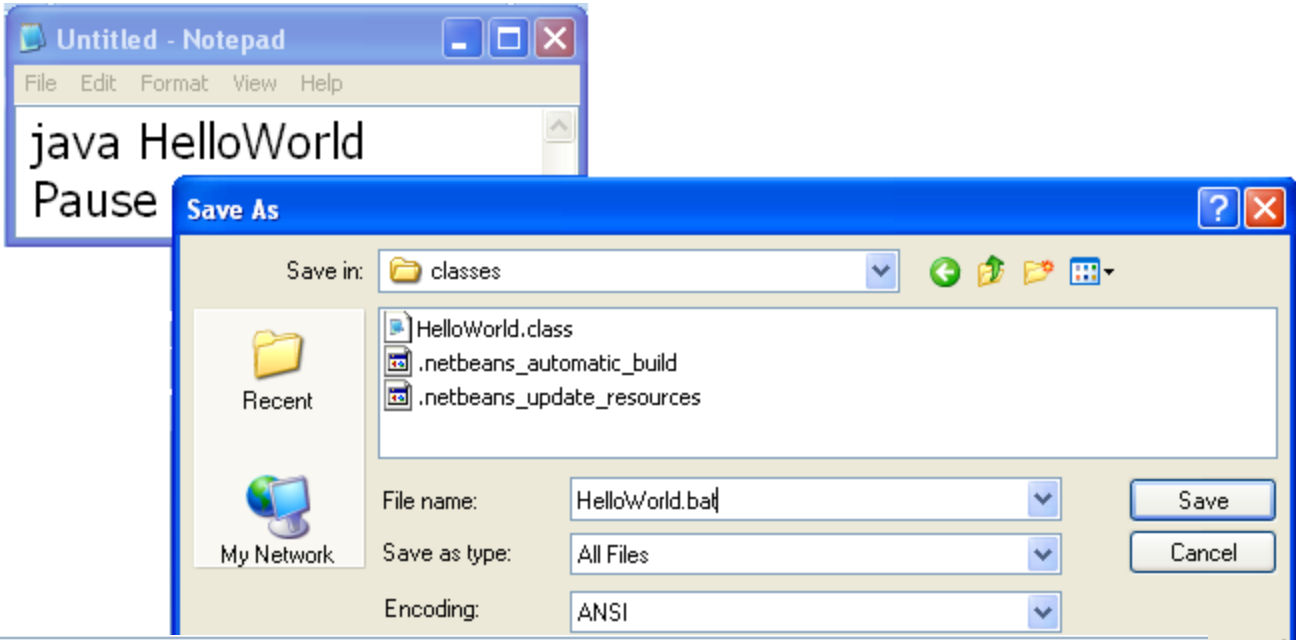


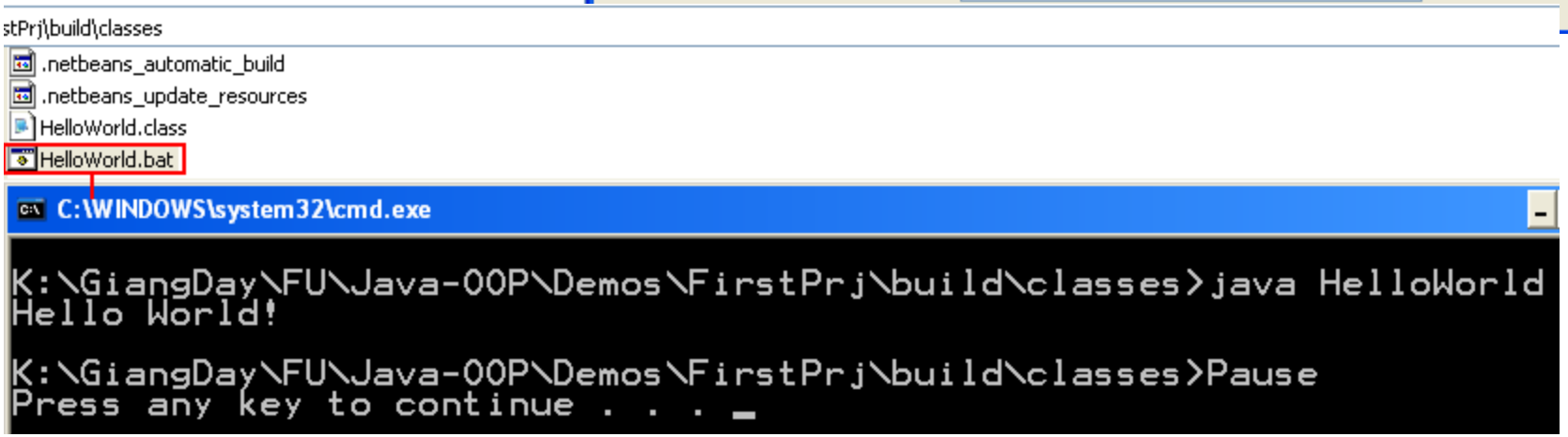


K: Change working drive to K
cd Change working directory

End users run Java Programs…

- Developer should support end users an easier way to run the program: **a BAT file**





Explain JDK and its tools

- javac (Java compiler)

```
javac [option] source
```

where,

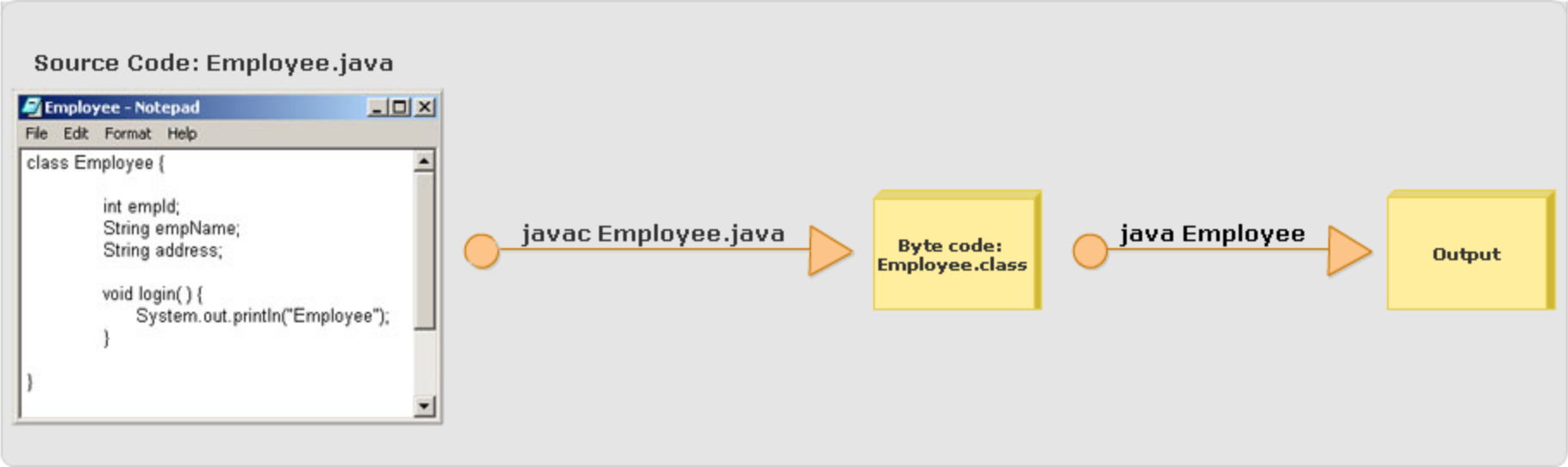
`source` is one or more file names that end with the extension .java.

- java (Java interpreter)

```
java [option] classname [arguments]
```

where,

`classname` is the name of the class file.



A Closer Look at the "Hello World!" Application

- ***Comments***
  - Traditional **/*this is a comment*/**
  - Comment to line end **//this is an end of line comment**
- ***Class declaration***
  - **public class ClassName { ... }**
  - For example: public class HelloWord { ... }
- ***The main Method – <u>Entry point</u> of Java program***
  - **public static void main(String[] args) {..}**
  - public and static can be written in either order
  - **The main method accepts a single argument: an array of elements of type String. A demonstration for passing strings to the main method will be presented in the next session.**

Common Problems (and Their Solutions)

- **Compiler Problems**

  'javac' is not recognized as an internal or external command, operable program or batch file

  ->Updating the PATH variable in the JDK

  - Syntax Errors (All Platforms)
  - Semantic Errors

- **Runtime Problems**

  - Exception in thread "main" java.lang.NoClassDefFoundError
  - Could not find or load main class HelloWorld.class

Classname is incorrect

Try and Explore

Change	To – If no error, try run it
public class HelloWorld	public class HelloWorld2
public class HelloWorld	class HelloWorld2
public static void main(String[] args)	public static void main(String args[])
public static void main(String[] args)	public void main(String[] args)
public static void main(String[] args)	void main(String[] args)

Summary

- An overview of Java technology as a whole.
- What to download, what to install, and what to type, for creating a simple "Hello World!" application.
- Discusses the "Hello World!" application.
- Trouble compiling or running the programs.