FPT UNIVERSITY

ĐẠI HỌC FPT CẦN THƠ

Session 01
Getting Started

Objectives

1. The Java Technology Phenomenon.
2. The "Hello World!" Application.
3. A Closer Look at "Hello World!“.
4. Common Problems (and Their Solutions).

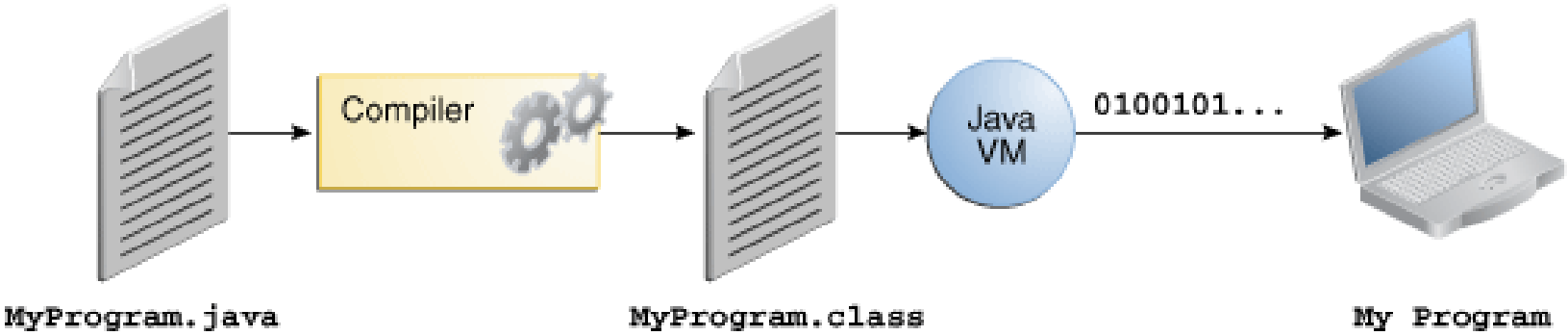
FPT UNIVERSITY

Objectives

1. **The Java Technology Phenomenon.**
2. The "Hello World!" Application.
3. A Closer Look at "Hello World!".
4. Common Problems (and Their Solutions).

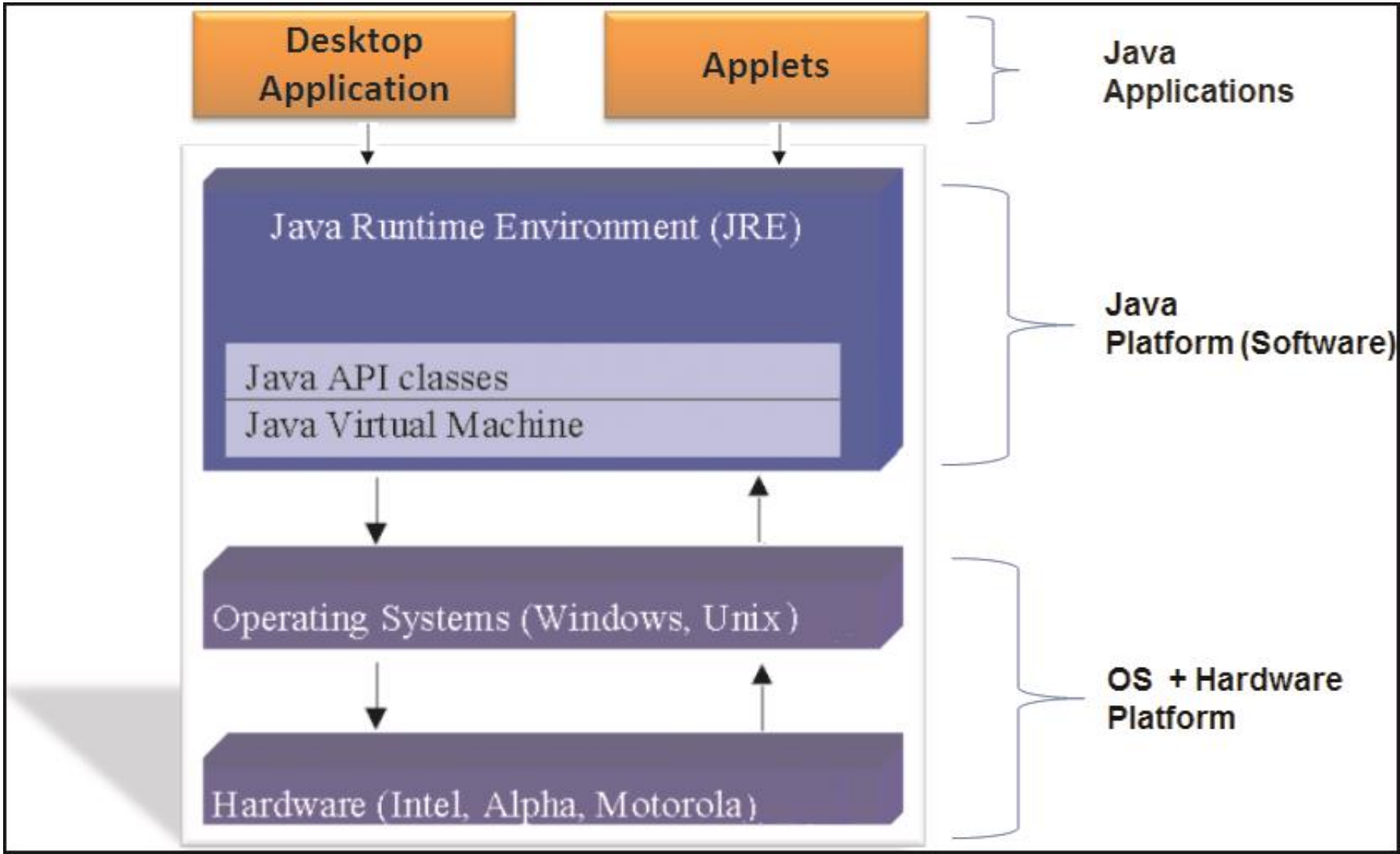
Java Technology

- The Java Programming Language.
  - High-level language.
  - All source code is first written in plain text files ending with the .java extension



Java Technology

- Java Platform.



Java Technology

- Java Platform:

- Is a software-only platform that runs on top of other hardware-based platforms.
- Contains **Java Runtime Environment (JRE)** with components namely:
  - Java Virtual Machine (JVM)
  - Java class library also referred to as Java Programming Interface (Java API)

Java Virtual Machine

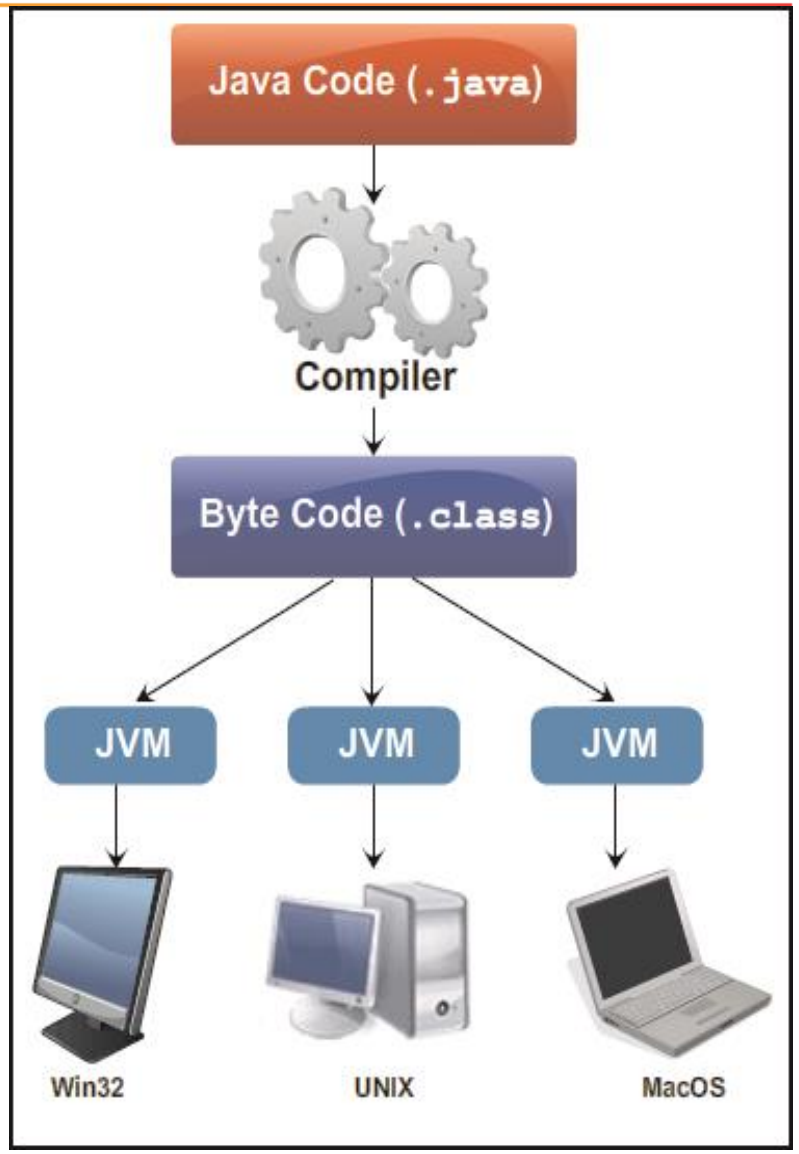
It is an executable engine that creates an environment for **executing Java compiled code**, that is, **bytecode**.

It is known as a virtual machine because it is an imitation of a **Java processor** on the **physical machine**.

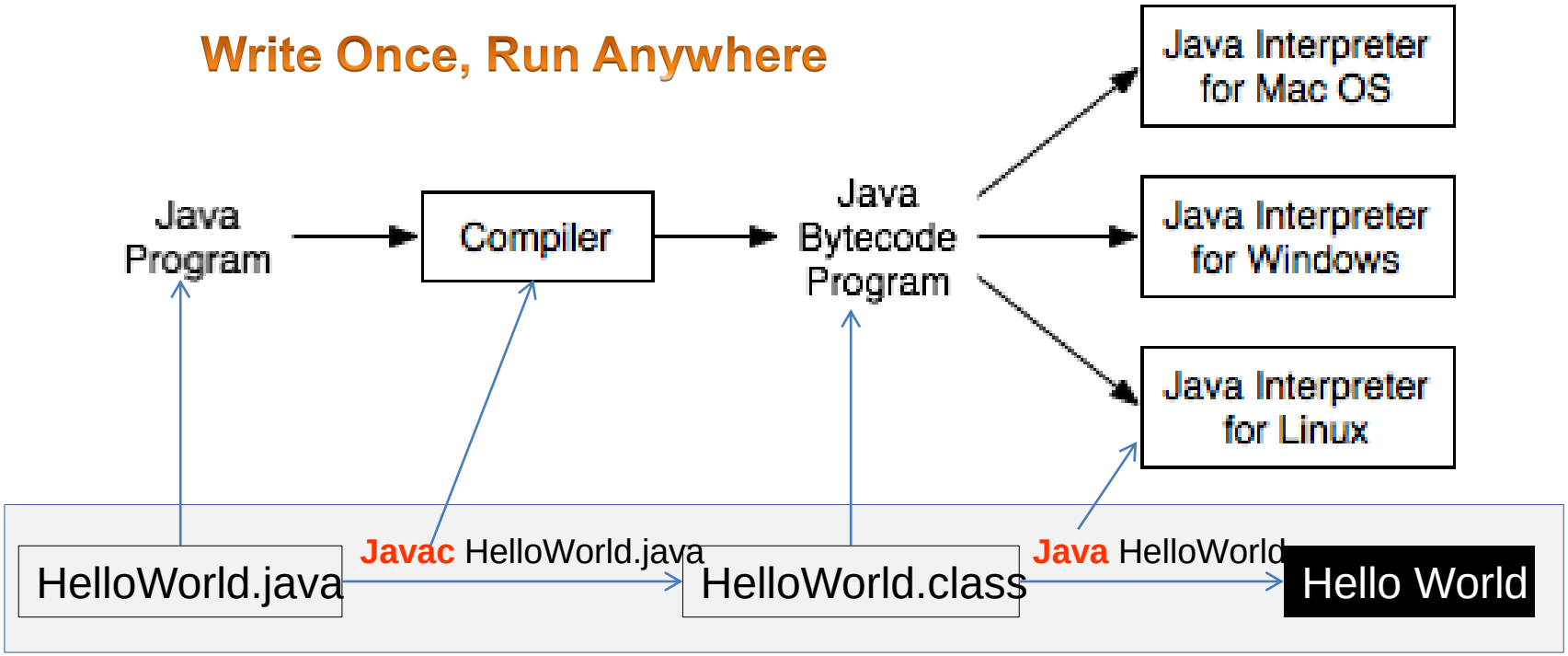
There are **different implementations of JVM** available for **different platforms**, such as Windows, Unix, and Solaris.

Java Virtual Machine

- **Bytecode**:
  - Is an intermediate form closer to machine representation.
  - Is an optimized set of instructions executed by the Java runtime environment.
  - This environment is known as JVM.
- The **same bytecode** can be executed by **different implementations** of JVM on **various platforms**.



Bytecode


Write Once, Run Anywhere
Java Program
Compiler
Java Bytecode Program
Java Interpreter for Mac OS
Java Interpreter for Windows
Java Interpreter for Linux
HelloWorld.java
Javac HelloWorld.java
HelloWorld.class
Java HelloWorld
Hello World

Java API

- It is a large collection of ready-made software components.
- These components are classes and interface grouped into libraries referred to as packages in Java.
- Example:
  - The **Swing** library provides classes for User Interface (UI) components.
  - The **Input/Output (I/O)** library provides the standard interface for reading and writing data into files stored in the system.

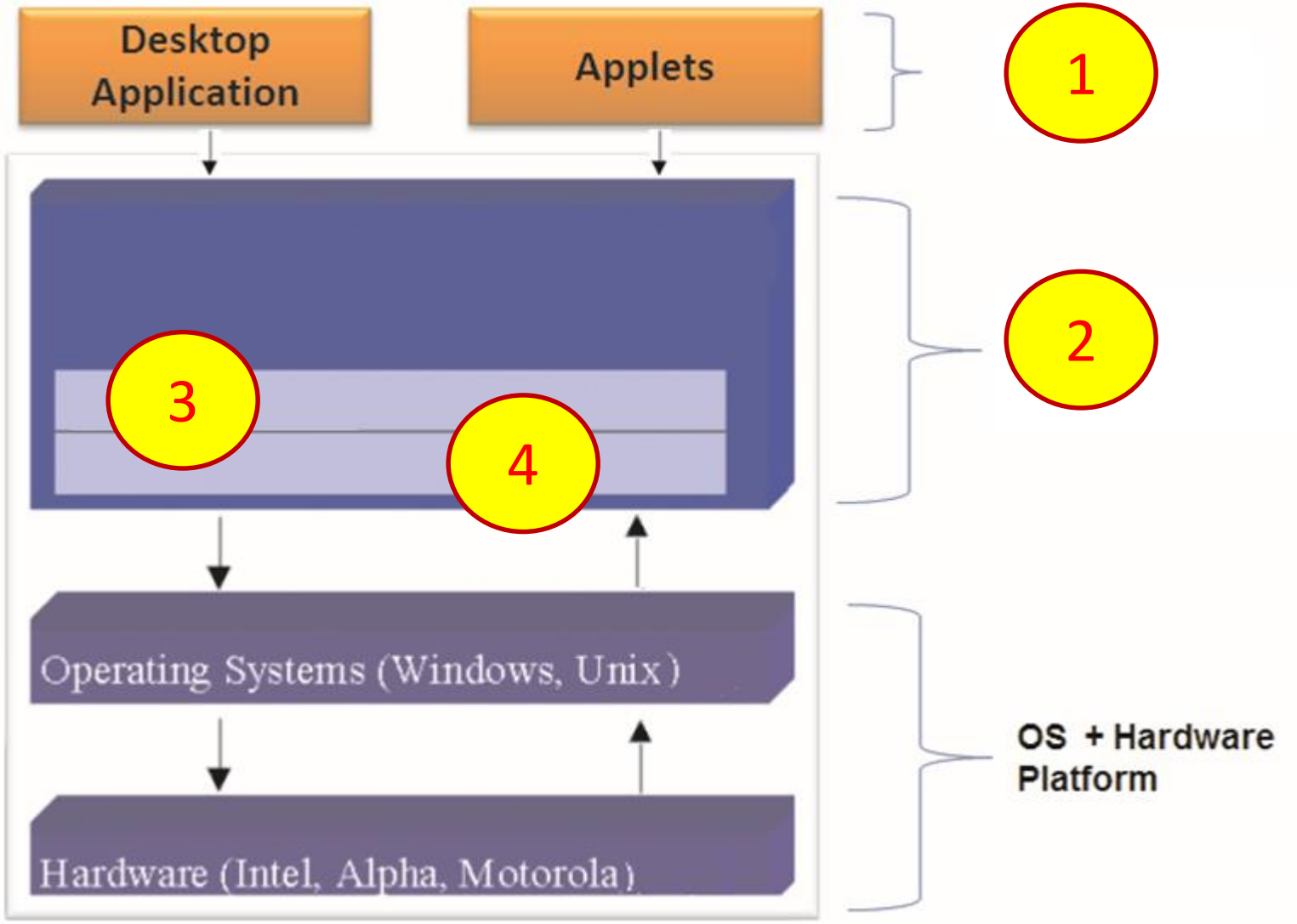
Objectives

1. **The Java Technology Phenomenon.**
2. The "Hello World!" Application.
3. A Closer Look at "Hello World!".
4. Common Problems (and Their Solutions).

Terms

1. Java Compiler
2. Java Applications
3. Java Platform
4. Byte Code
5. Java Runtime Environment
6. Application / Java App
7. Java Programming Interface (Java API)
8. Java Virtual Machine (JVM)

Terms


Desktop Application
Applets
1
2
3
4
Operating Systems (Windows, Unix)
Hardware (Intel, Alpha, Motorola)
OS + Hardware Platform

Objectives

1. The Java Technology Phenomenon.
2. **The "Hello World!" Application.**
3. A Closer Look at "Hello World!".
4. Common Problems (and Their Solutions).

The "Hello World!" Application

To write your first program, you'll need:

- The Java SE Development Kit.
- The NetBeans IDE.

Steps to create your first application

- Create an IDE project.
- Add code to the generated source file.
- Compile the source file into a .class file.
- Run the program.

Introduction JDK

- Explain JDK and its tools
  - javac (Java compiler)
  - java (Java interpreter)
- Configure JDK
  - Path
    - The Path variable is set to point to the location of Java executables (javac.exe, java.exe).
  - Classpath
    - Classpath is an environment variable that specifies the location of the class files and libraries needed for the Java compiler (javac) to compile applications.

Explain JDK and its tools

- javac (Java compiler)

```
javac [option] source
```

where,

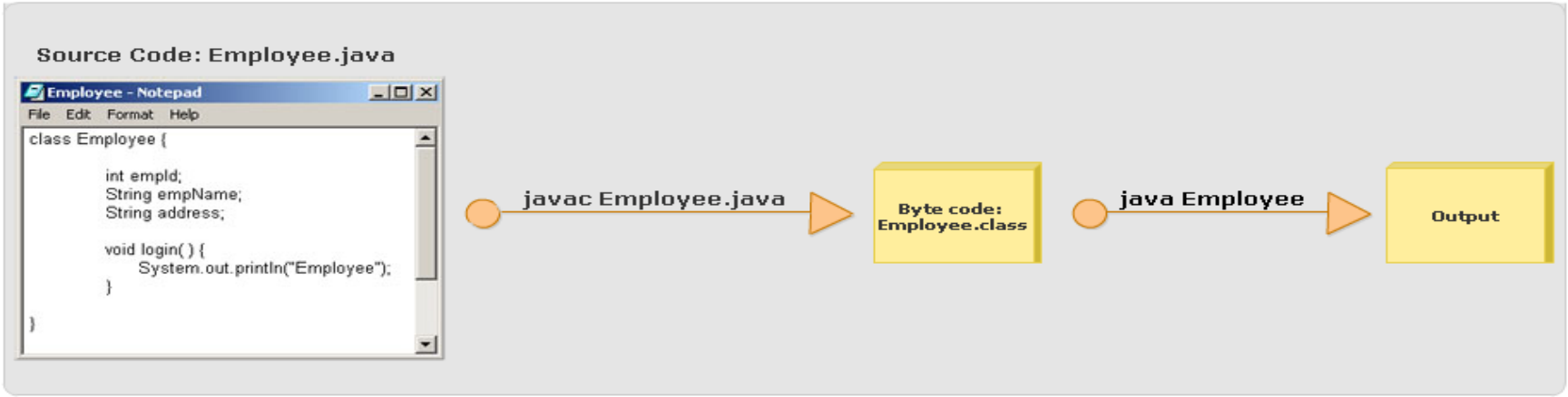
`source` is one or more file names that end with the extension .java.

- java (Java interpreter)

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

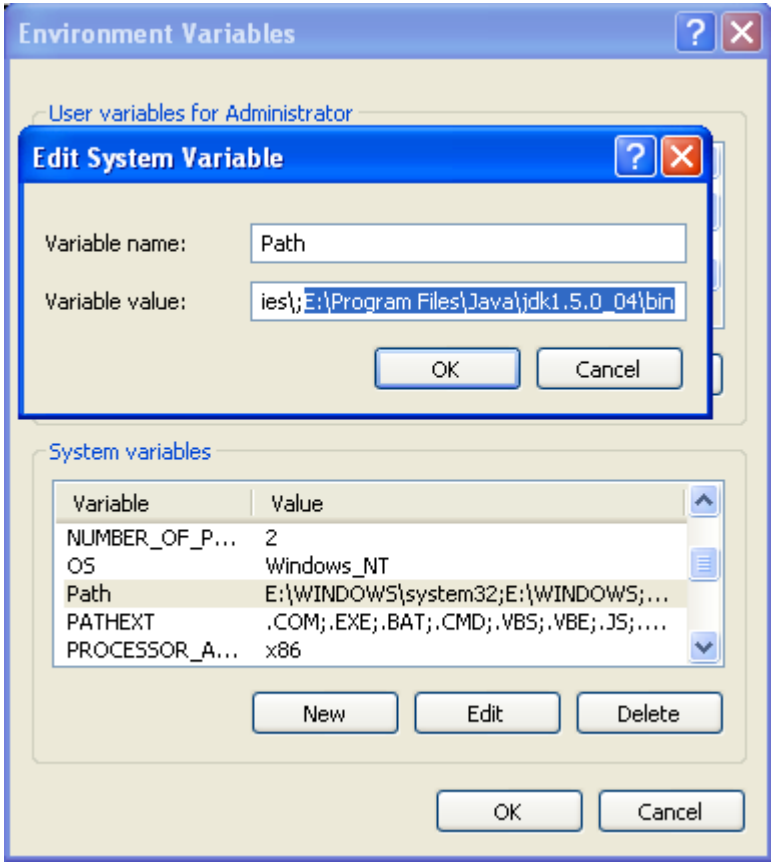
where,

`classname` is the name of the class file.

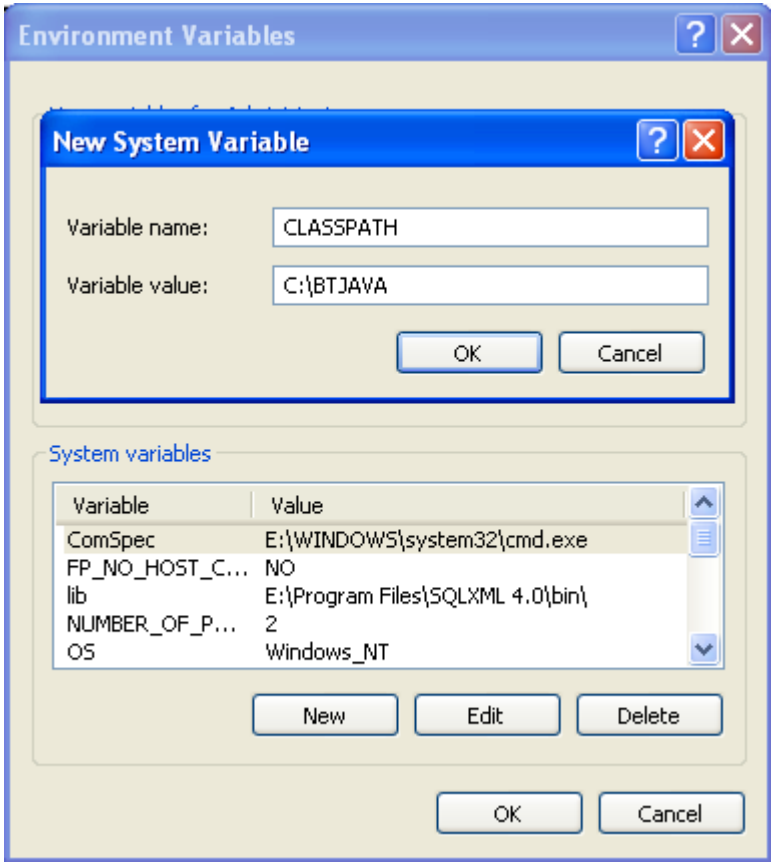


Configure JDK

Path:

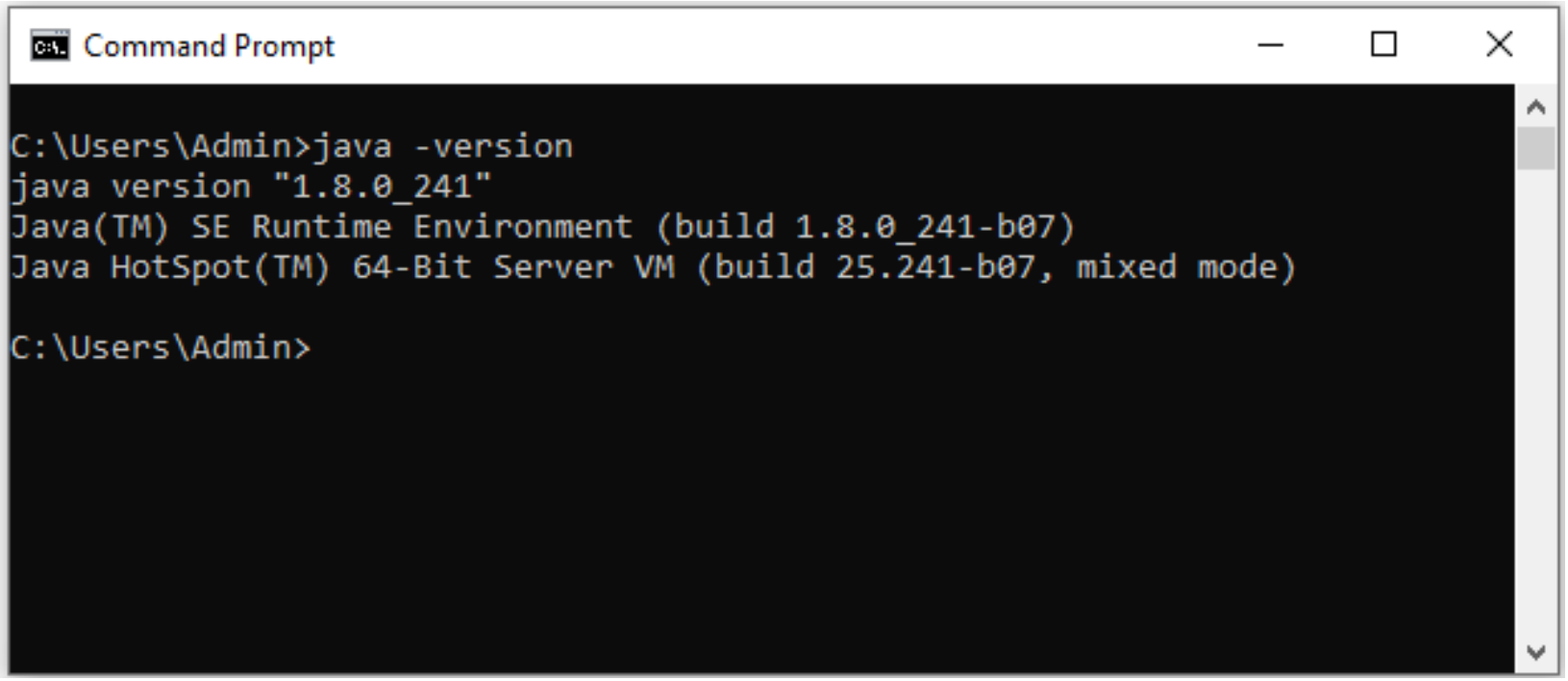


Classpath



Test Configure JDK

- Button Windows  R or type CMD on toolbar search.
- Write: java –version



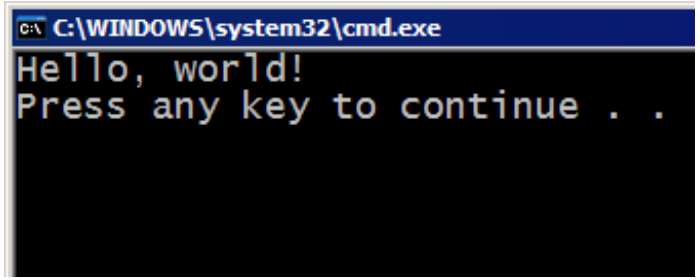
"Hello World!" for Microsoft Windows

- To write your first program, you'll need:
  - The Java SE Development Kit (JDK)
  - A text editor
- Steps to Create Your First Application
  - Create a source file
  - Compile the source file into a .class file
  - Run the program

A basic Java program (1)

```
public class Hello {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

- **code** or **source code**: The sequence of instructions in a program.
  - The code in this program instructs the computer to display a message of **Hello, world!** on the screen.
- **output**: The messages printed to the user by a program.
- **console**: The text box onto which output is printed.
  - Some editors pop up the console as an external window, and others contain their own console window.



A basic Java program (2)

- Bring up a shell, or "command," window.
  - Change your current directory to the directory where your file is located.
  - Type the following command and press Enter

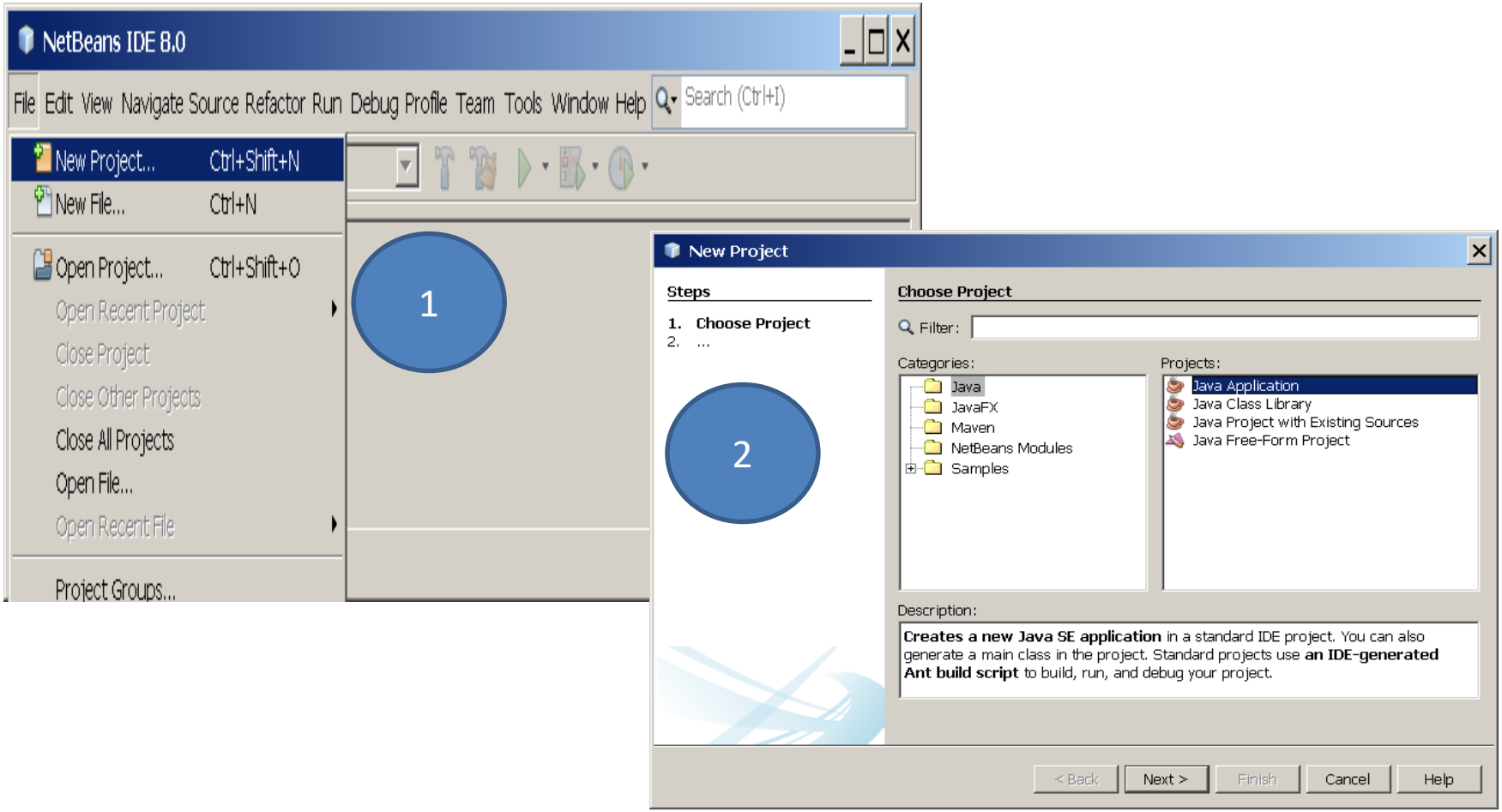
```
javac Hello.java
```

- Run this program
  - In the same directory, enter the following command at the prompt:

```
java Hello
```

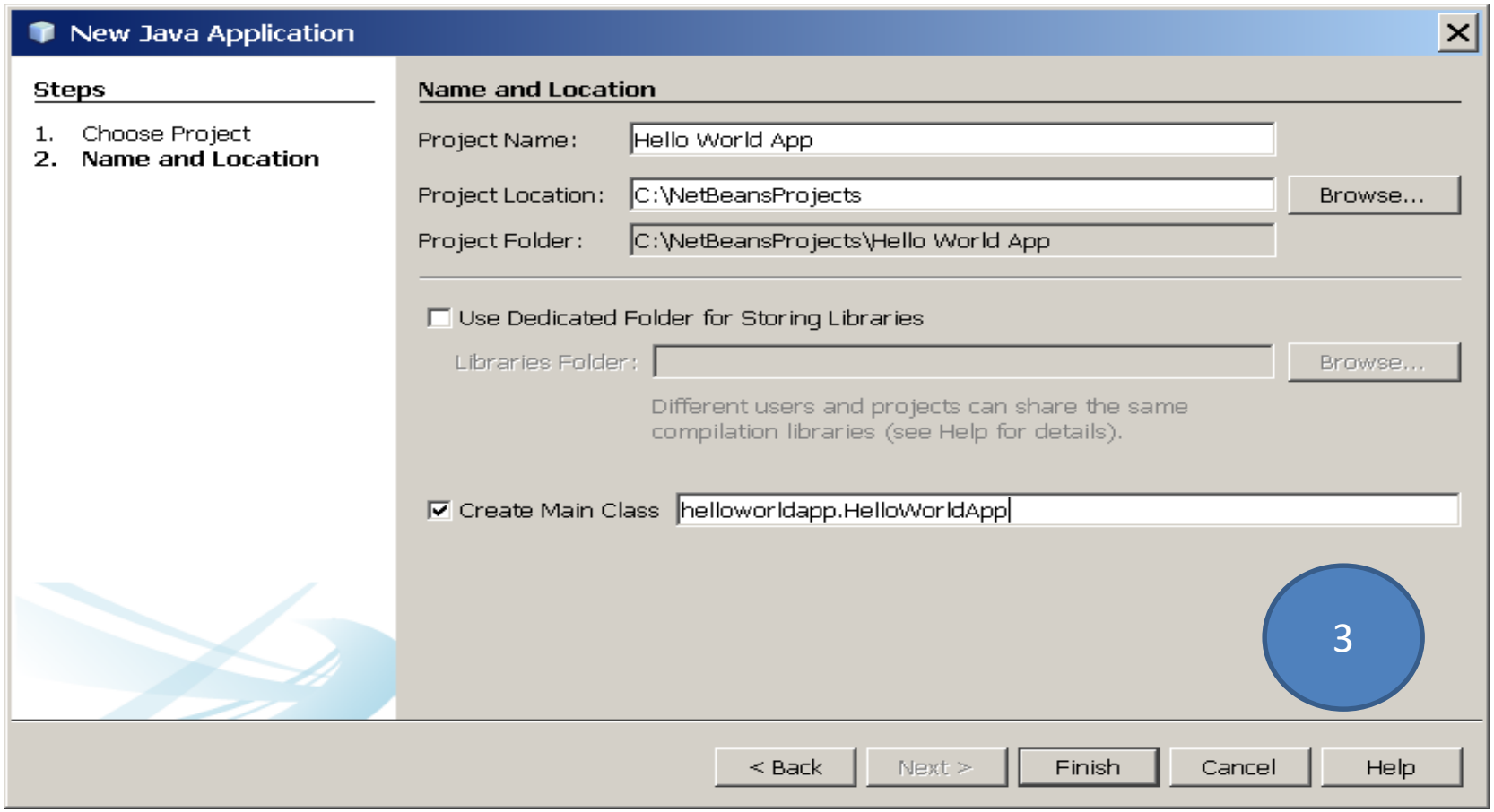
Create an IDE project(1)

- "Hello World!" for the NetBeans IDE



Create an IDE project(2)

- Put the name for project and class



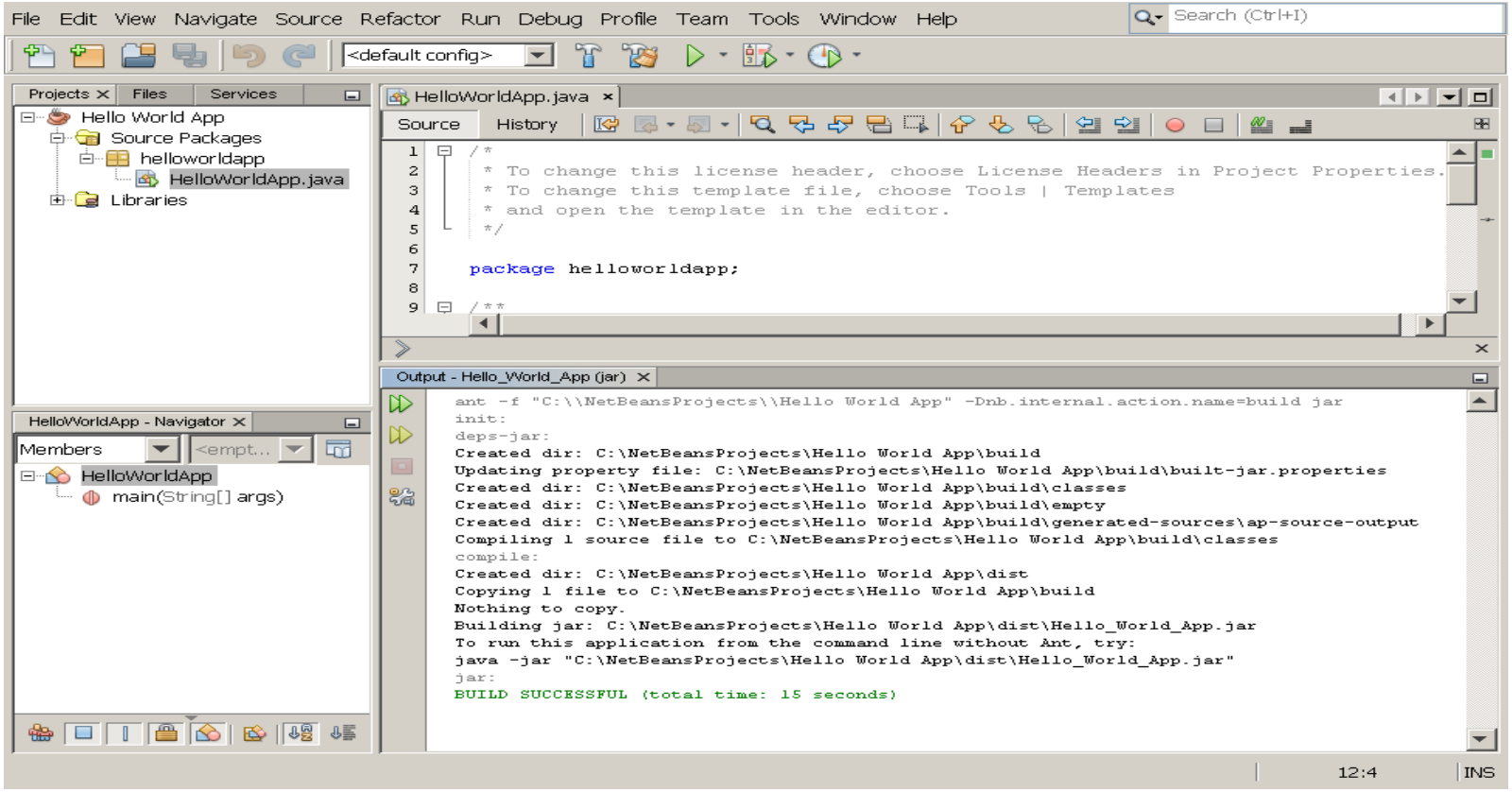
Add code to the generated source file

- Add the below code to main function

```
System.out.println("Hello World!");
  // Display the string.
```

Compile the source file into a .class file

- Run\Build project



Objectives

1. The Java Technology Phenomenon.
2. **The "Hello World!" Application.**
3. A Closer Look at "Hello World!".
4. Common Problems (and Their Solutions).

Terms

1. Java Programming Language
2. Java Code
3. Java Compiler / JavaC
4. Java interpreter
5. Java SE Development Kit (JDK).
6. NetBeans IDE
7. Path
8. Environment Variables

Objectives

1. The Java Technology Phenomenon.
2. The "Hello World!" Application.
3. **A Closer Look at "Hello World!".**
4. Common Problems (and Their Solutions).

A Closer Look at the "Hello World!" Application

- Comments
  - Traditional /*this is a comment*/
  - End-of-line //this is an end of line comment
  - Java document (java doc comments)
- Class declaration
  - public class ClassName { ... }
  - For example: public class Hello{ ... }
- The main Method
  - 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.

Structure of Java programs

```
public class <name> {
    public static void main(String[] args) {
        <statement>;
        <statement>;
        ...
        <statement>;
    }
}
```

- Every executable Java program consists of a **class**
  - that contains a **method** named `main`
    - that contains the **statements** (commands) to be executed

Java terminology

- **class**: A module that can contain executable code.
  - Every program you write will be a class.
- **statement**: An executable command to the computer.
- **method**: A named sequence of statements that can be executed together to perform a particular action.
  - A special method named `main` signifies the code that should be executed when your program runs.
  - Your program can have other methods in addition to `main`.

Syntax

- **syntax**: The set of legal structures and commands that can be used in a particular programming language.

- some Java syntax:
  - every basic Java statement ends with a semicolon `;`
  - The contents of a class or method occur between `{` and `}`

Objectives

1. The Java Technology Phenomenon.
2. The "Hello World!" Application.
3. **A Closer Look at "Hello World!".**
4. Common Problems (and Their Solutions).

Terms

1. Comment
2. Declaration
3. Method
4. Statement
5. Syntax

Terms

```
8  /**
9   *
10  * @author Admin
11  */
12 public class HelloWorld {
13
14     /**
15      * @param args the command line arguments
16      */
17     public static void main(String[] args) {
18         // TODO code application logic here
19         String st = "Hello World!";
20         System.out.println(st);
21     }
22 }
```

Objectives

1. The Java Technology Phenomenon.
2. The "Hello World!" Application.
3. A Closer Look at "Hello World!“.
4. **Common Problems (and Their Solutions)**

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: HelloWorldApp
  - Could not find or load main class HelloWorldApp.class

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.

ĐẠI HỌC FPT CẦN THƠ

cantho.fpt.edu.vn

Student's task

- Install Netbean.
- Run HelloWorld Project.