

Lecture 10

Java – How to create and Execute a Java application?

Mostafa Elhosseini

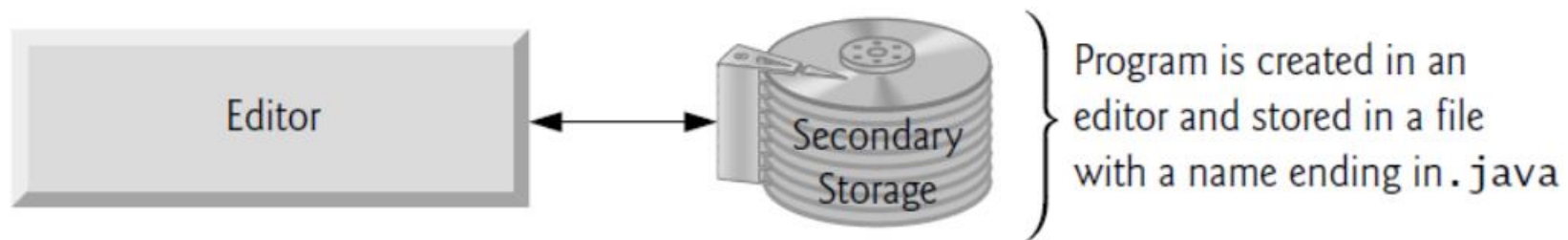
Professor at College of Computer Science and
Engineering

Taibah University

<https://youtube.com/drmelhosseini>

| Create a program

- Editing a file with an **editor program**, normally known simply as an **editor**.
- Using the editor, you type a Java program (**source code**),
- make any necessary corrections and **save it** on a secondary storage device, such as your hard drive.
- **Java source code** files are given a name ending with the **.java** extension, **indicating** that the **file contains Java source code**



| Create a program

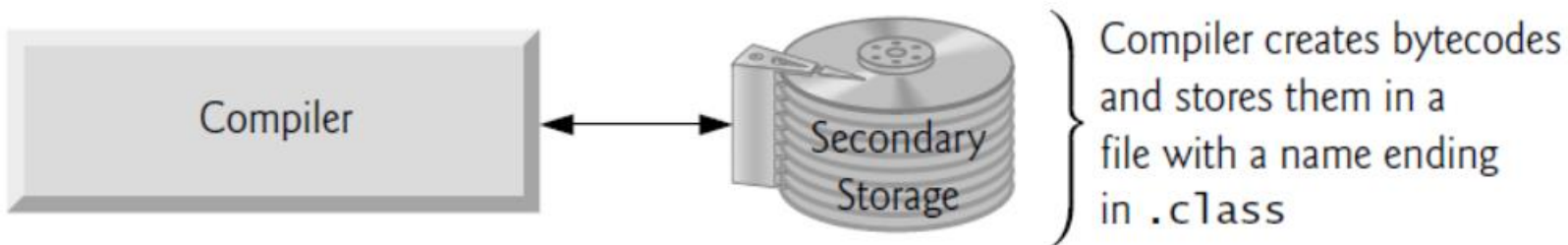
- Two editors widely used on **Linux** systems are **vi** and **emacs** (). **Windows** provides **Notepad**. **macOS** provides **TextEdit**.
- Many freeware and shareware editors are also available online, including **Notepad++** (<http://notepad-plus-plus.org>), **EditPlus** (<http://www.editplus.com>), **TextPad** (<http://www.textpad.com>), **jEdit** (<http://www.jedit.org>) and more.

| Create a program

- Integrated development environments (**IDEs**) provide tools that support the software development process, such as **editors**, **debuggers** for locating logic errors that cause programs to execute incorrectly and **more**.
- The most popular Java IDEs are
 - Eclipse (<http://www.eclipse.org>)
 - IntelliJ IDEA (<http://www.jetbrains.com>)
 - NetBeans (<http://www.netbeans.org>)

| Compile a Java Program into a Bytecodes

- Use the command **javac** (the **Java compiler**) to compile a program
 - **Javac Welcome.java** (Command Prompt in Windows)
- If the program compiles, the compiler produces a **.class** file called **Welcome.class**.
- **IDEs** typically provide a **menu** item, such as **Build** or **Make**, that **invokes the javac command** for you



| Compile a Java Program into a Bytecodes

- When using javac, **if you receive a message** such as:
 - **bad command or filename**
 - **javac: command not found**
 - **javac is not recognized as an internal or external command,**
- then your Java software installation was not completed properly. This indicates that the system's **PATH environment variable** was not set properly

| Compile a Java Program into a Bytecodes

- The **Java compiler** translates **Java source code** into **bytecodes** that represent the tasks to execute in the execution phase

| Java Virtual Machine

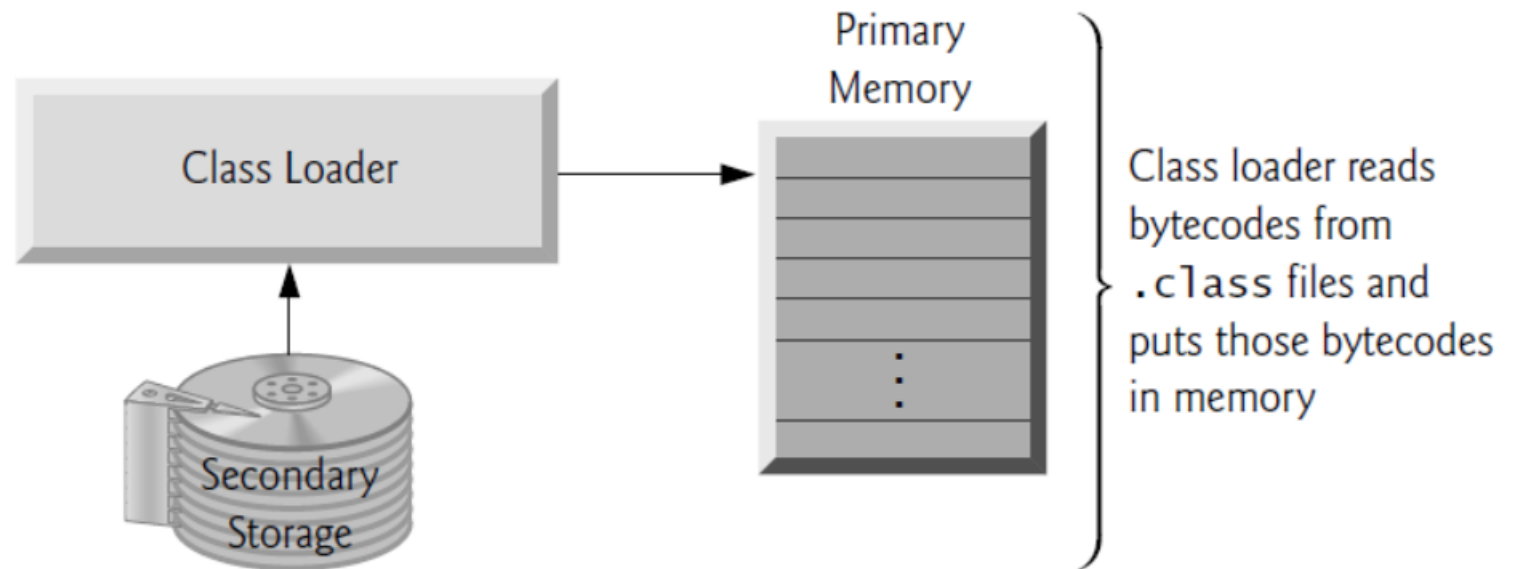
- The **Java Virtual Machine (JVM)**—a part of the **JDK** and the foundation of the Java platform—**executes bytecodes**.
- A virtual machine (VM) is a software application that **simulates a computer** but hides the underlying operating system and hardware from the programs that interact with it.
- If the **same VM is implemented on many computer platforms**, **applications written for that type of VM** can be **used** on all those **platforms**.
- The **JVM** is one of the **most widely used virtual machines**

| Java Virtual Machine

- **bytecode instructions are platform independent.** So, Java's bytecodes are **portable**—without recompiling the source code, the **same bytecode instructions can execute on any platform** containing a JVM that understands the version of Java in which the bytecodes were compiled.
- The **JVM** is invoked by the java command
 - **java Welcome**
- IDEs typically provide a menu item, such as Run, that invokes the java command for you

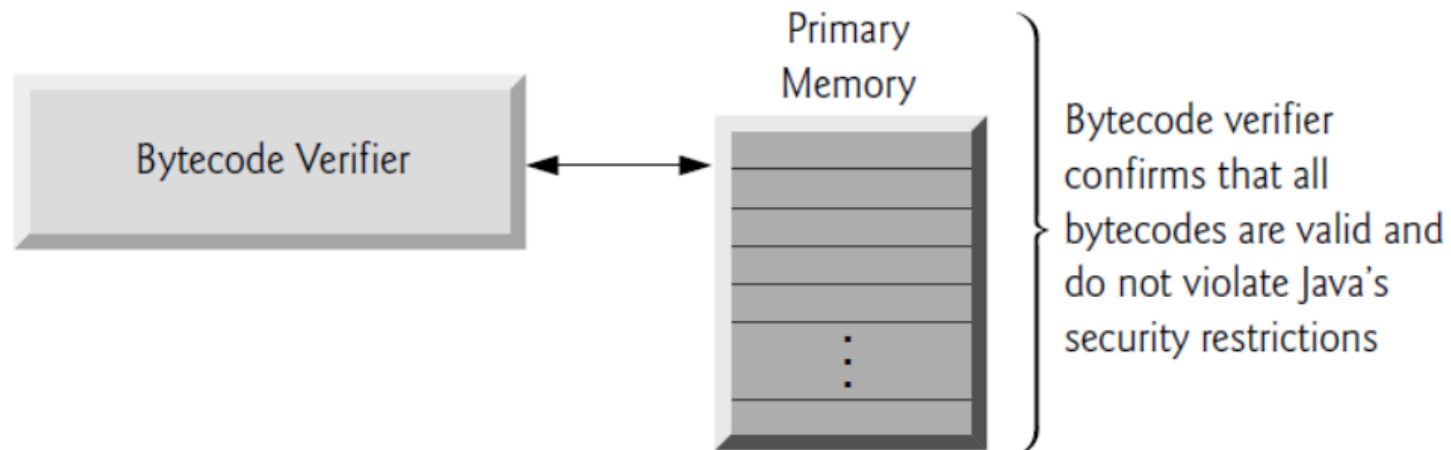
| Loading a program into memory

- the **JVM** places the program in memory to execute it—this is known as loading .The **JVM's class loader** takes the **.class files** containing the program's **bytecodes** and transfers them to **primary memory**
- It also **loads any of the .class files** provided by Java that your program uses



| Bytecode Verification

- the bytecode verifier examines their bytecodes to ensure that they're valid and do not violate Java's security restrictions.
- Java enforces strong security to make sure that Java programs arriving over the network do not damage your files or your system (as computer viruses and worms might).



| Execution

- the JVM executes the bytecodes to perform the program's specified actions
- In **early Java versions**, the JVM was simply a **Java-bytecode interpreter**. Most programs would execute slowly, because the JVM would interpret and execute one bytecode at a time
- **Today's JVMs** typically execute bytecodes using a **combination of interpretation and just-intime (JIT) compilation**

| Execution

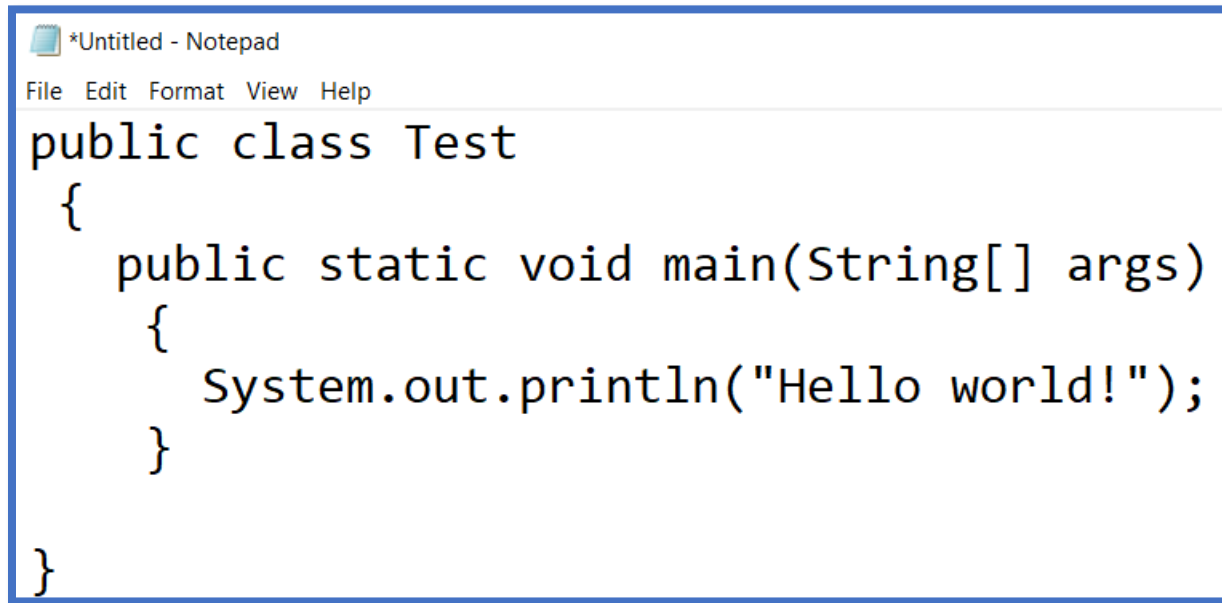
- In this process, the **JVM analyzes the bytecodes** as they're interpreted, searching for **hot spots**—bytecodes that execute frequently.
- For these parts, a **just-in-time (JIT) compiler**, such as Oracle's Java HotSpot™ compiler, translates the bytecodes into the computer's machine language.
- When the JVM encounters these compiled parts again, the faster machine-language code executes

| Problems That May Occur at Execution Time

- Errors such as **division by zero** occur as a program runs, so they're called **runtime errors** or **execution-time errors**.
- **Fatal runtime errors** cause programs to terminate immediately without having successfully performed their jobs.
- **Nonfatal runtime errors** allow programs to run to completion, often producing incorrect results

| Hello world Example

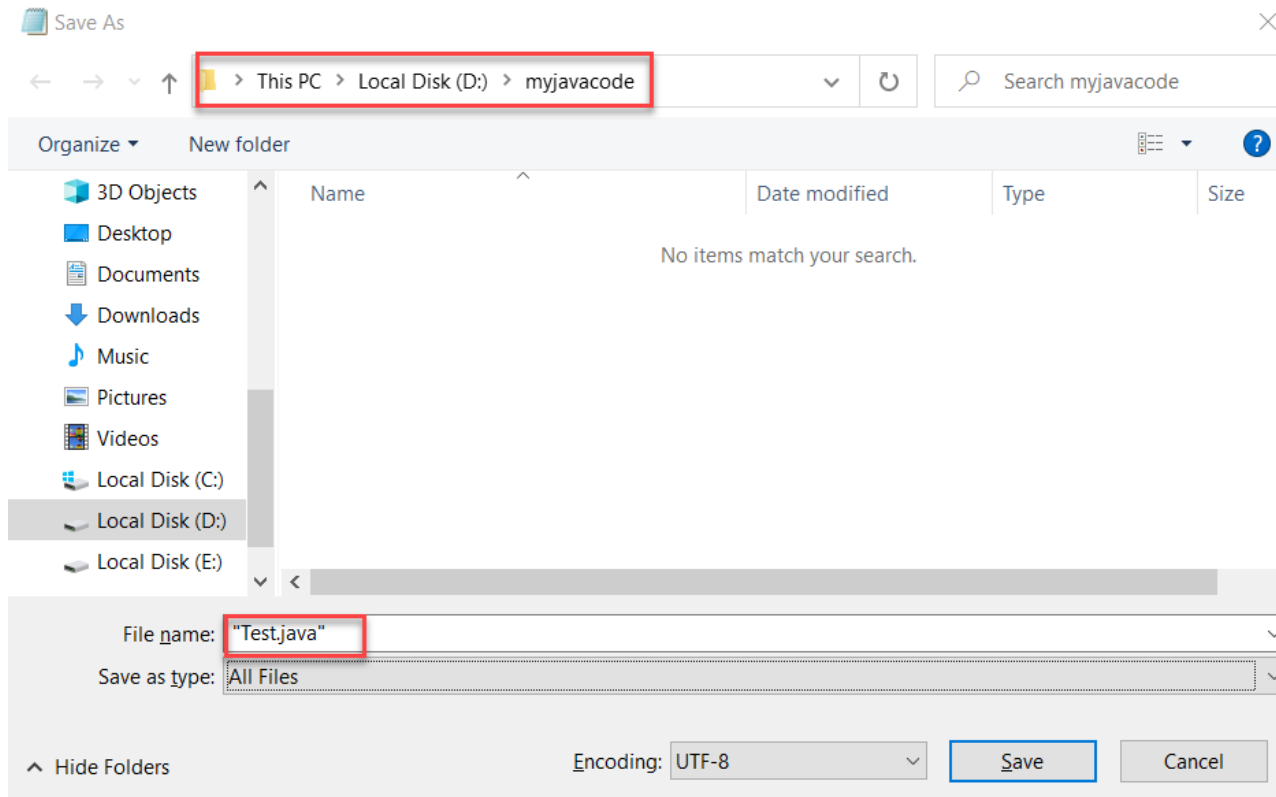
- Open Editor - Notepad

A screenshot of a Notepad window titled '*Untitled - Notepad'. The window has a menu bar with 'File', 'Edit', 'Format', 'View', and 'Help'. The text area contains the following Java code:

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

| Hello world Example

■ Save File- Notepad



| Hello world Example

- Error – Compiling File

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19043.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\melho>d:

D:\>cd \myjavacode

D:\myjavacode>javac Test.java
Test.java:1: error: class HelloWorld is public, should be declared in a file named HelloWorld.java
public class HelloWorld
      ^
1 error
```

| Hello world Example

- Class name = file name

 Test.java - Notepad

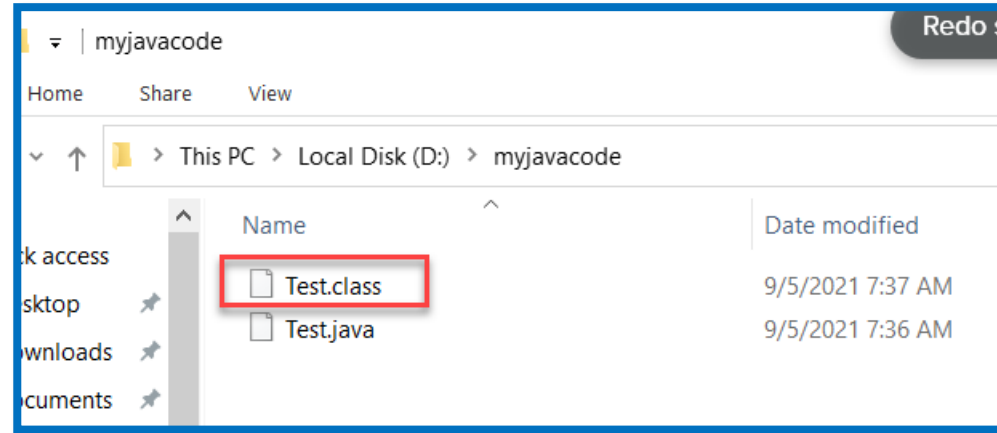
File Edit Format View Help

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

| Hello world Example

- Compile again

```
D:\myjavacode>javac Test.java  
D:\myjavacode>
```



| Hello world Example

- Execute

```
D:\myjavacode>javac Test.java
```

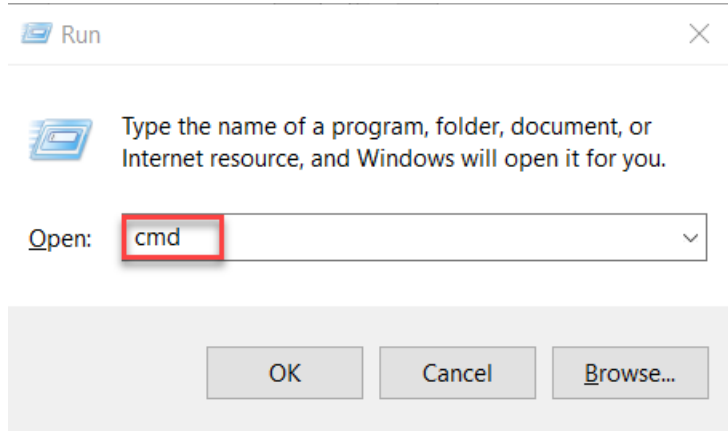
```
D:\myjavacode>java Test
```

```
Hello world!
```

```
D:\myjavacode>
```

| Hello world Example

- Open command windows



| References

- Java: How to Program, 9th Edition (Deitel) 9th Edition by Paul Deitel , Harvey M. Deitel
- Head First Java, 2nd Edition 2nd Edition by Kathy Sierra, Bert Bates
- Think Java: How to Think Like a Computer Scientist 1st Edition by Allen B. Downey, Chris Mayfield