Core Java Programming

What is JAVA

- Java is a high-level object-oriented programming language developed by the Sun Microsystems.
- Father of java programming is James gosling.
- Initiation year -1991.
- JDK 1.0 was released on January 23, 1996.
- It is platform Independent Language means java program are capable to run in any platform like windows, Linux, Unix etc.

Java History

- **James Gosling**, **Mike Sheridan**, and **Patrick Naughton** initiated the Java language project in June 1991. The small team of sun engineers called **Green Team**.

- Originally designed for small, embedded systems in electronic appliances like set-top boxes TV,VCR etc.

- Firstly, it was called **"Greentalk"** by James Gosling and file extension was .gt.

- After that, it was called **Oak** and was developed as a part of the Green project.

Java

- In 1995, Oak was renamed as **"Java“.**
- Notice that Java is just a name not an acronym.
- Originally developed by James Gosling at Sun Microsystems.
- In 1995, Time magazine called **Java one of the Ten Best Products of 1995**.
- JDK 1.0 released in(January 23, 1996).

Why JAVA

- Java is easy to learn. Java was designed to be easy to use and is therefore easy to write, compile, debug, and learn than other programming languages.
- Java is object-oriented. This allows you to create modular programs and reusable code.
- Java is platform-independent. One of the most significant advantages of Java is its ability to move easily from one computer system to another.

Why JAVA

- Because of Java's robustness, ease of use, cross-platform capabilities and security features, it has become a language of choice for providing worldwide Internet solutions.

Why java

- There are many Application where java is currently used. Some of them are as follows:

A. Desktop Applications such as media player, antivirus etc.
B. Web Applications such as irctc.co.in, gmail.com etc.
C. Enterprise Applications such as banking applications.
D. Mobile Application
E. Embedded System (telecom, smart cards ,missiles and satellites , computer networking etc)
F. Robotics
G. Games etc.

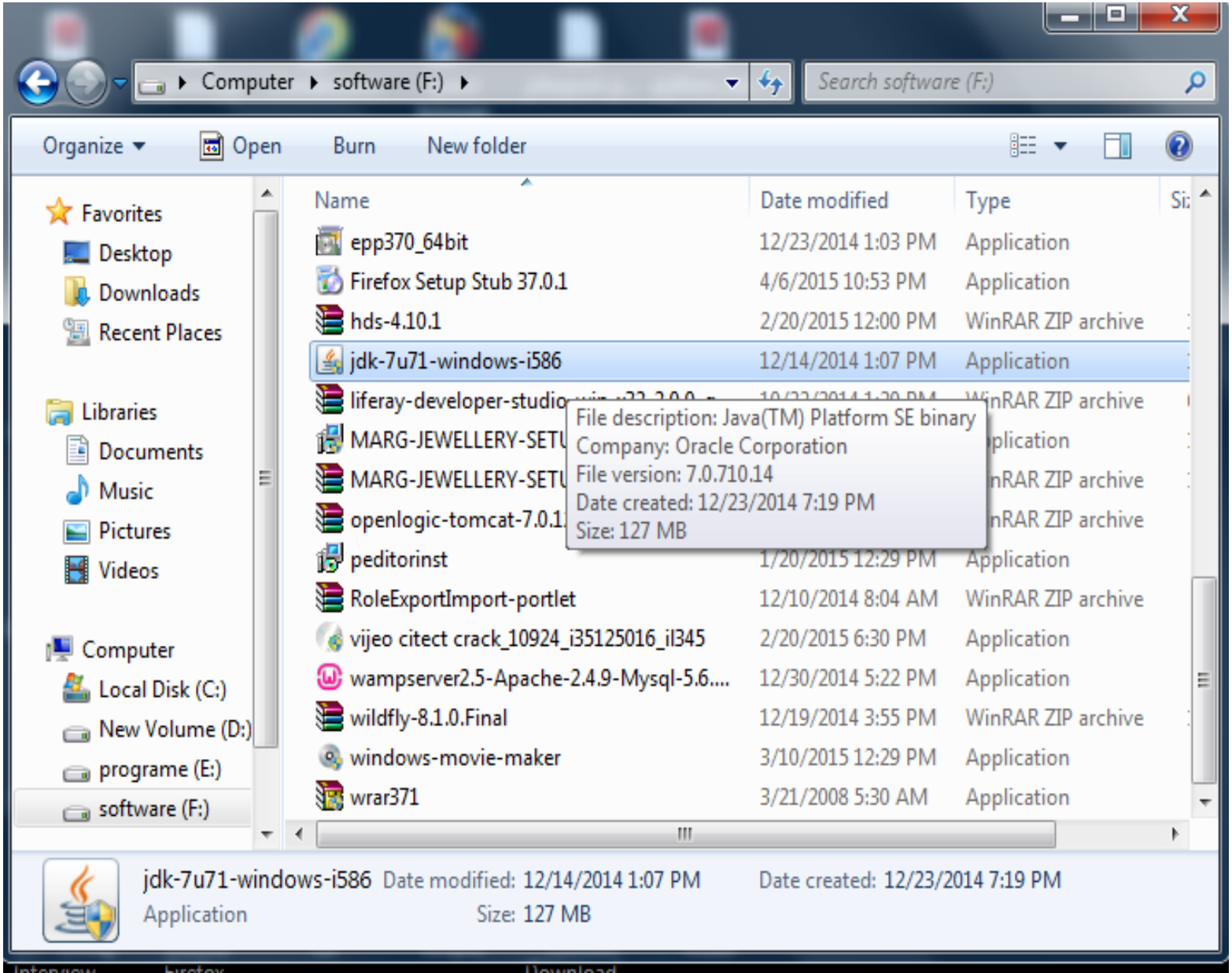
Features of java

- Features of Java

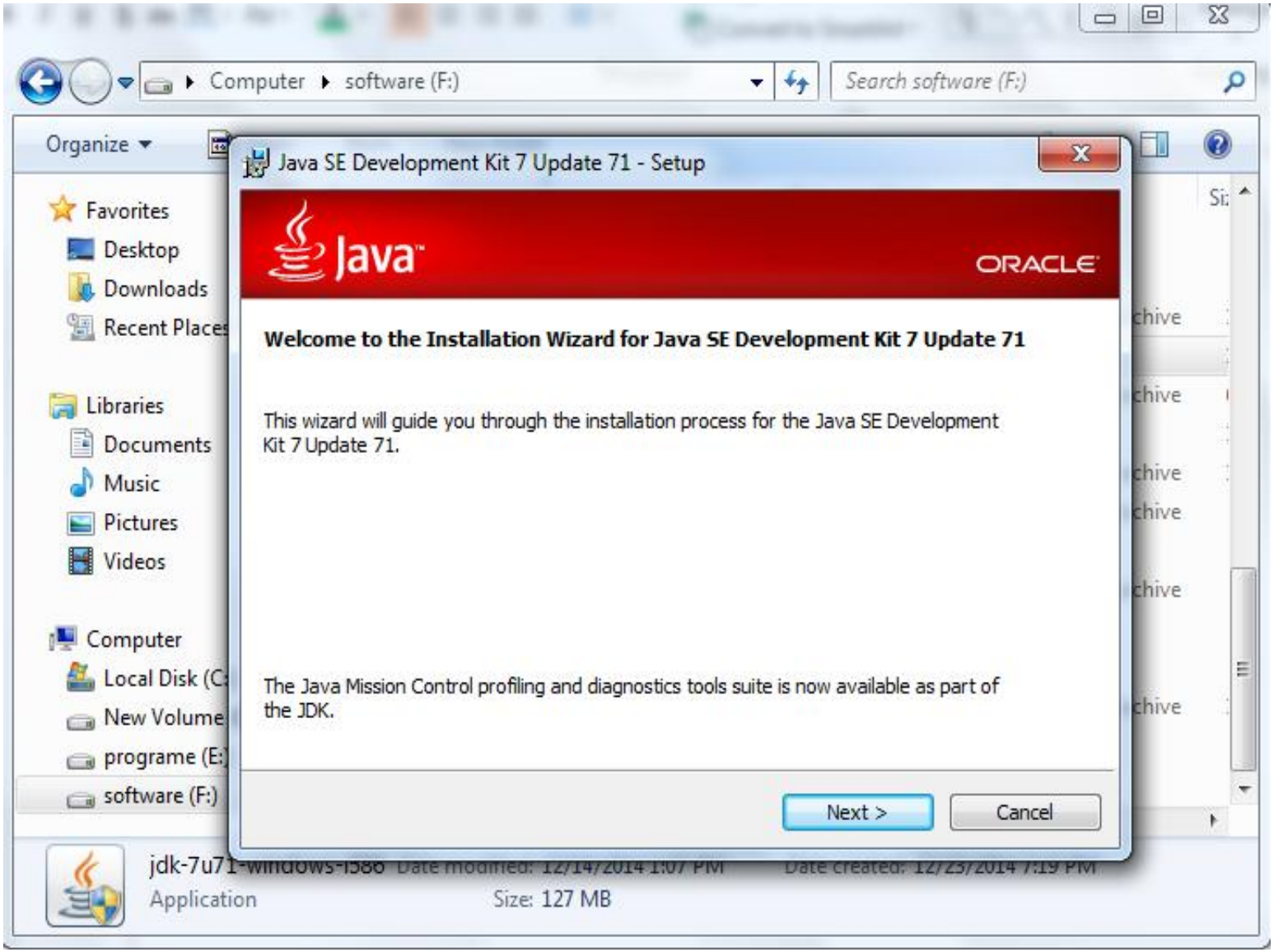
Installation of Java

1. First Of all you have a .exe file of Jdk1.7(if not Then Download from Oracle Site).
2. Now this exe file can be save anywhere (let us consider D:\ drive)
3. Open d:\ drive and double click on jdk1.7 exe file.
4. Press Next button until complete.
5. Now check the java where it is installed
6. If successfully installed java then go to the path where java is installed
7. C:\program File(X86)\java\jdk1.7

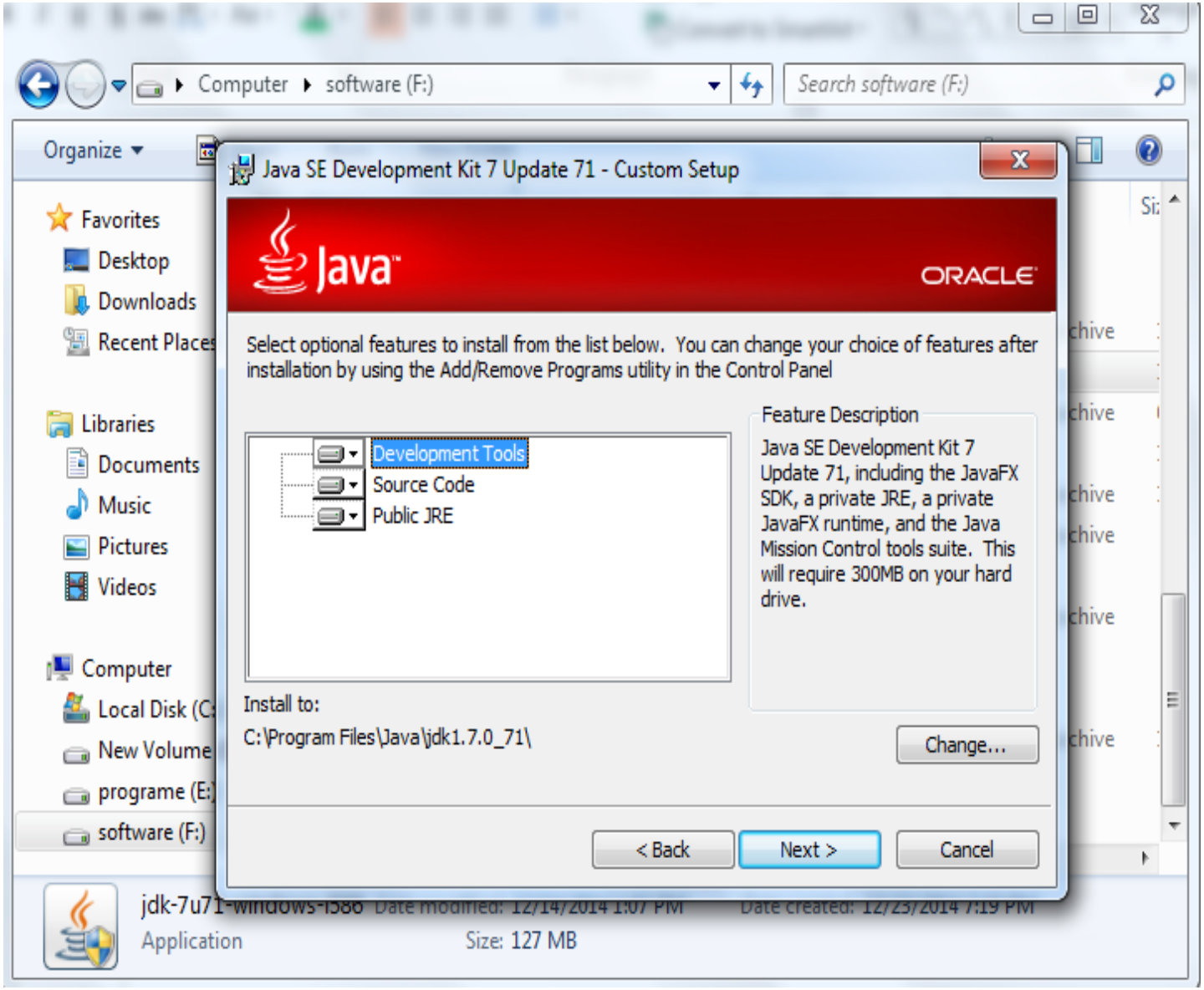
- Doble click on the java installation exe file

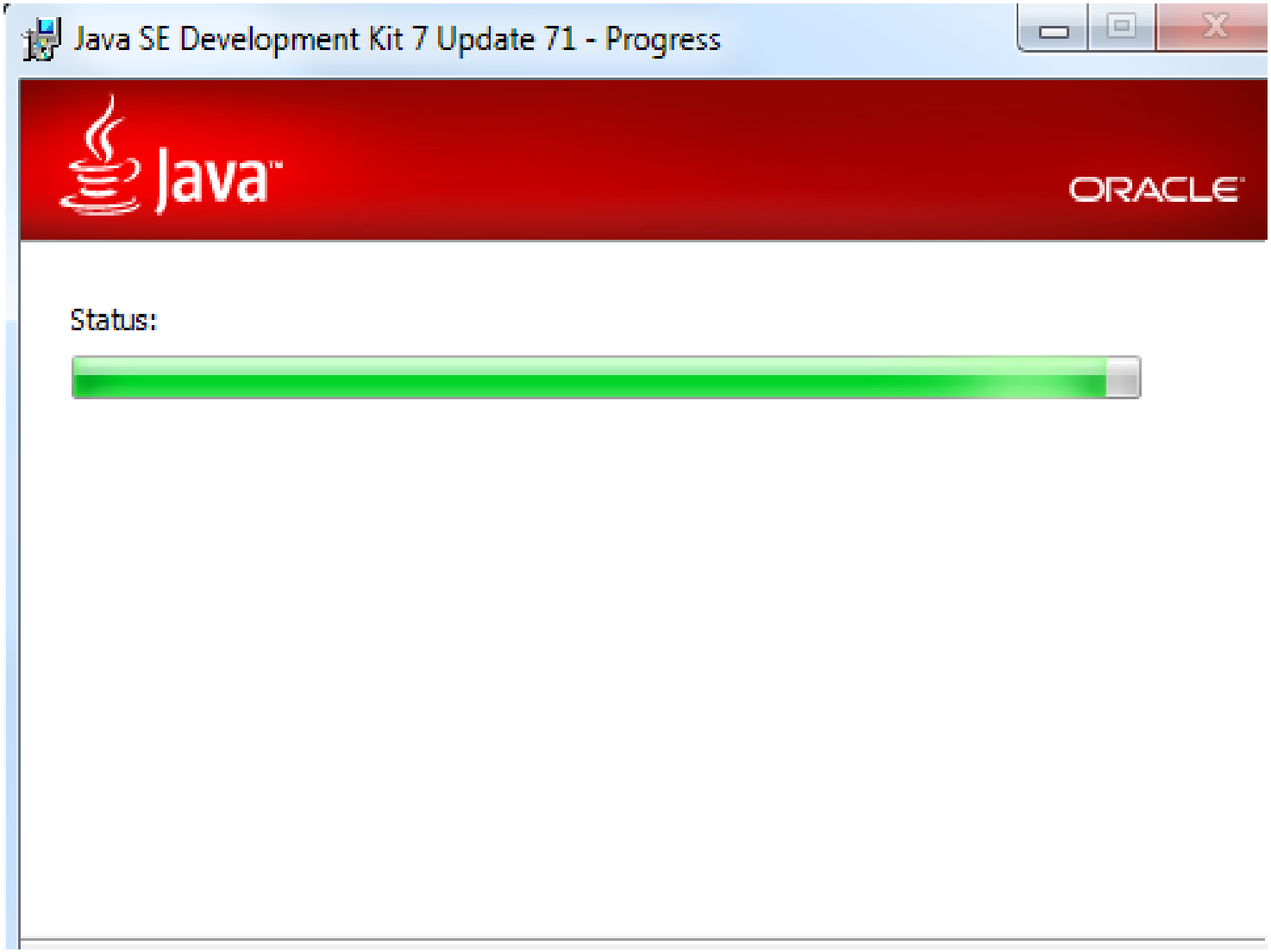


- Click Next Button

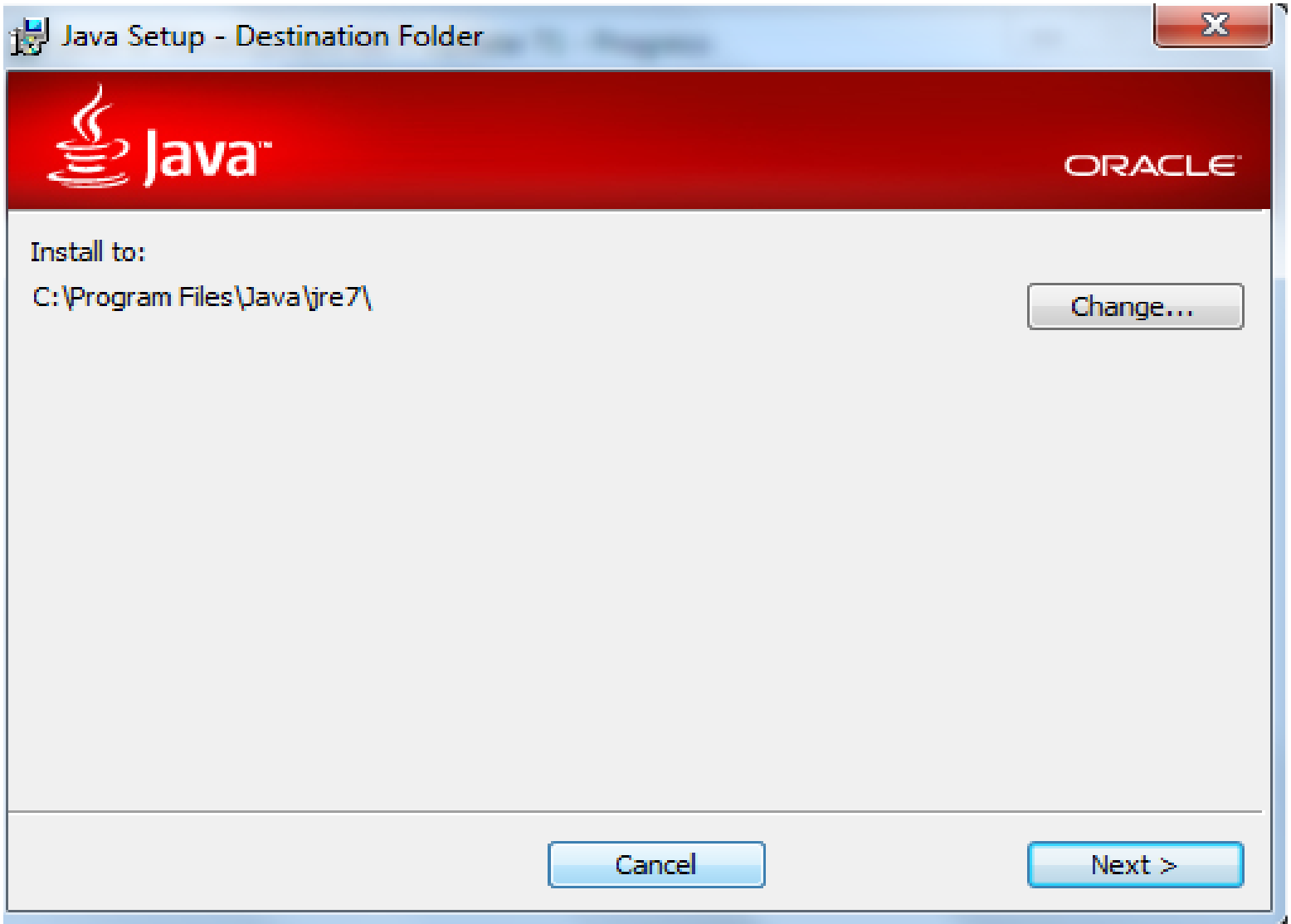


- Then again select Next Button




Java SE Development Kit 7 Update 71 - Progress
Java
ORACLE
Status:

- Cloick next button




Java Setup - Progress
Java
ORACLE
Status:
Registering Java Runtime Environment
3 Billion Devices Run Java
Computers, Printers, Routers, Cell Phones, BlackBerry, Kindle, Parking Meters, Public Transportation Passes, ATMs, Credit Cards, Home Security Systems, Cable Boxes, TVs...
ORACLE

- Now close the button press

C:\Program Files\Java

Search Java

Organize ▾ Include in library ▾ Share with ▾ Burn New folder

Videos

Computer

Local Disk (C:)

New Volume (D:)

programe (E:)

software (F:)

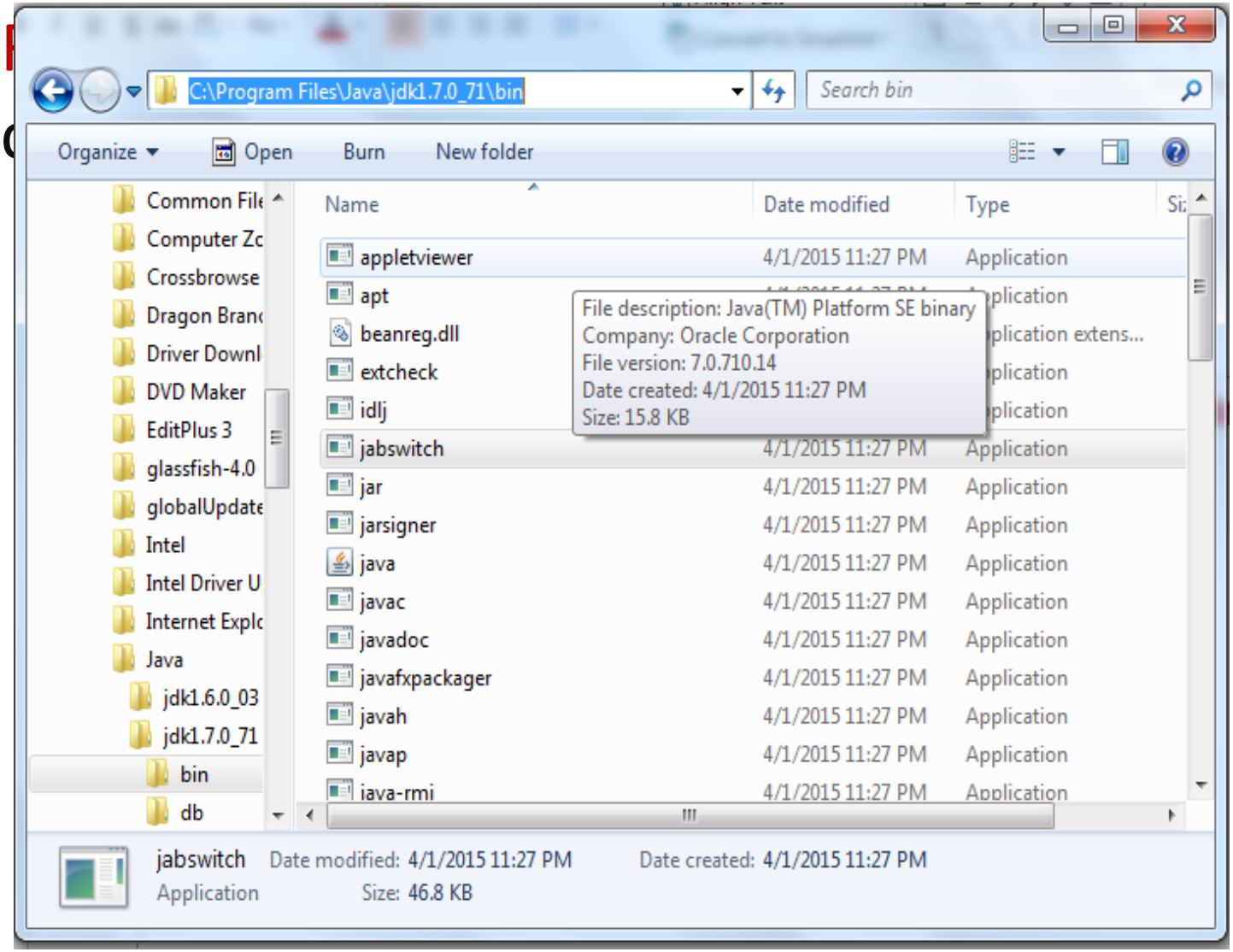
Name	Date modified	Type	Size
jdk1.6.0_03	3/23/2015 2:20 AM	File folder	
jdk1.7.0_71	4/1/2015 11:28 PM	File folder	
jre1.6.0_03	3/23/2015 2:20 AM	File folder	
jre7			

Date created: 4/1/2015 11:27 PM
Size: 281 MB
Folders: bin, db, include, jre, lib
Files: COPYRIGHT, LICENSE, README, release, src, ...

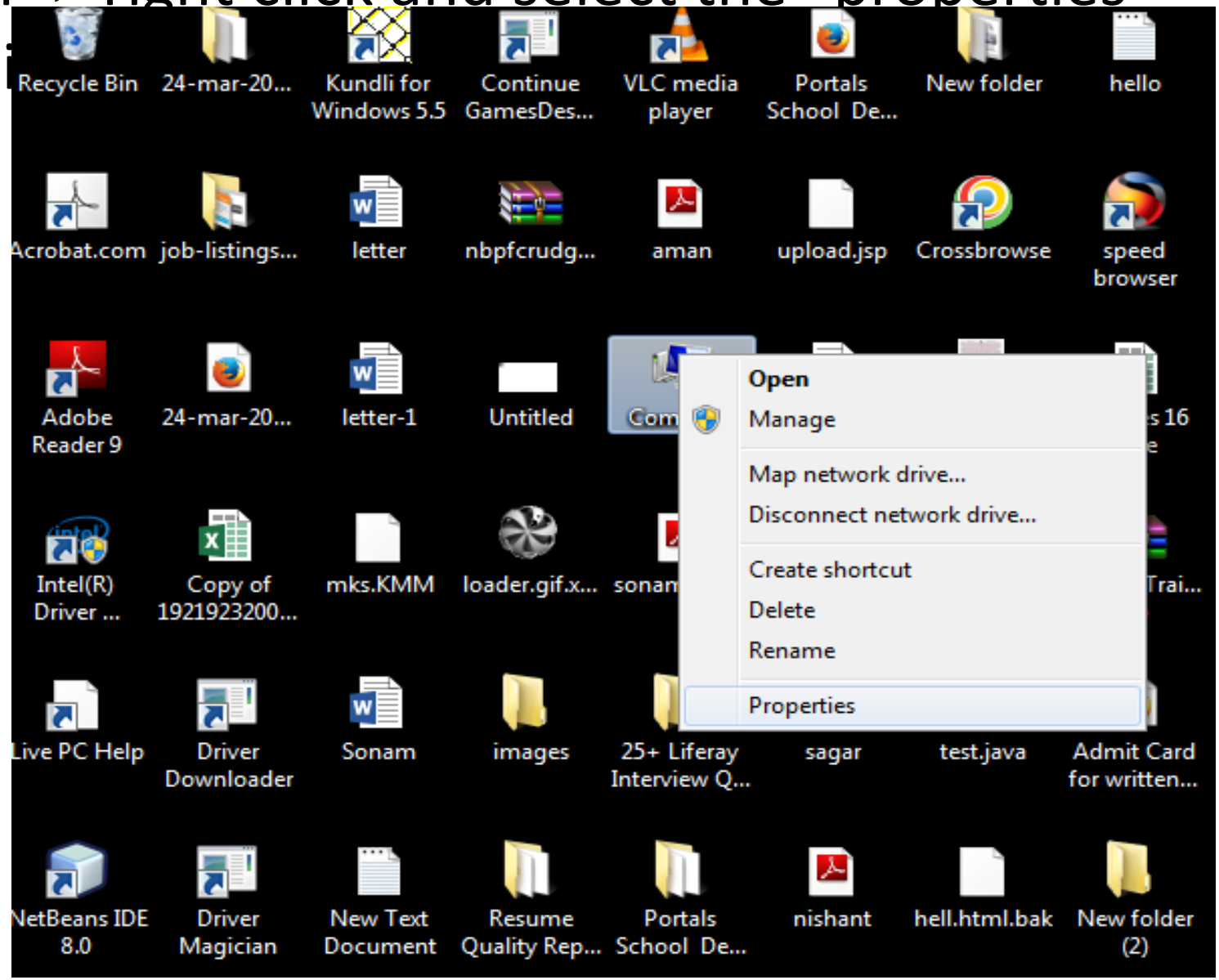
4 items

Set the Path of Java

1. Go to "C:\program



1. Open Desktop and select the “computer” icon -> right click and select the “properties Opti



Control Panel ▸ All Control Panel Items ▸ System

Search Control Panel

Control Panel Home

- Device Manager
- Remote settings
- System protection
- Advanced system settings

See also

- Action Center
- Windows Update
- Performance Information and Tools

View basic information about your computer

Windows edition

Windows 7 Ultimate

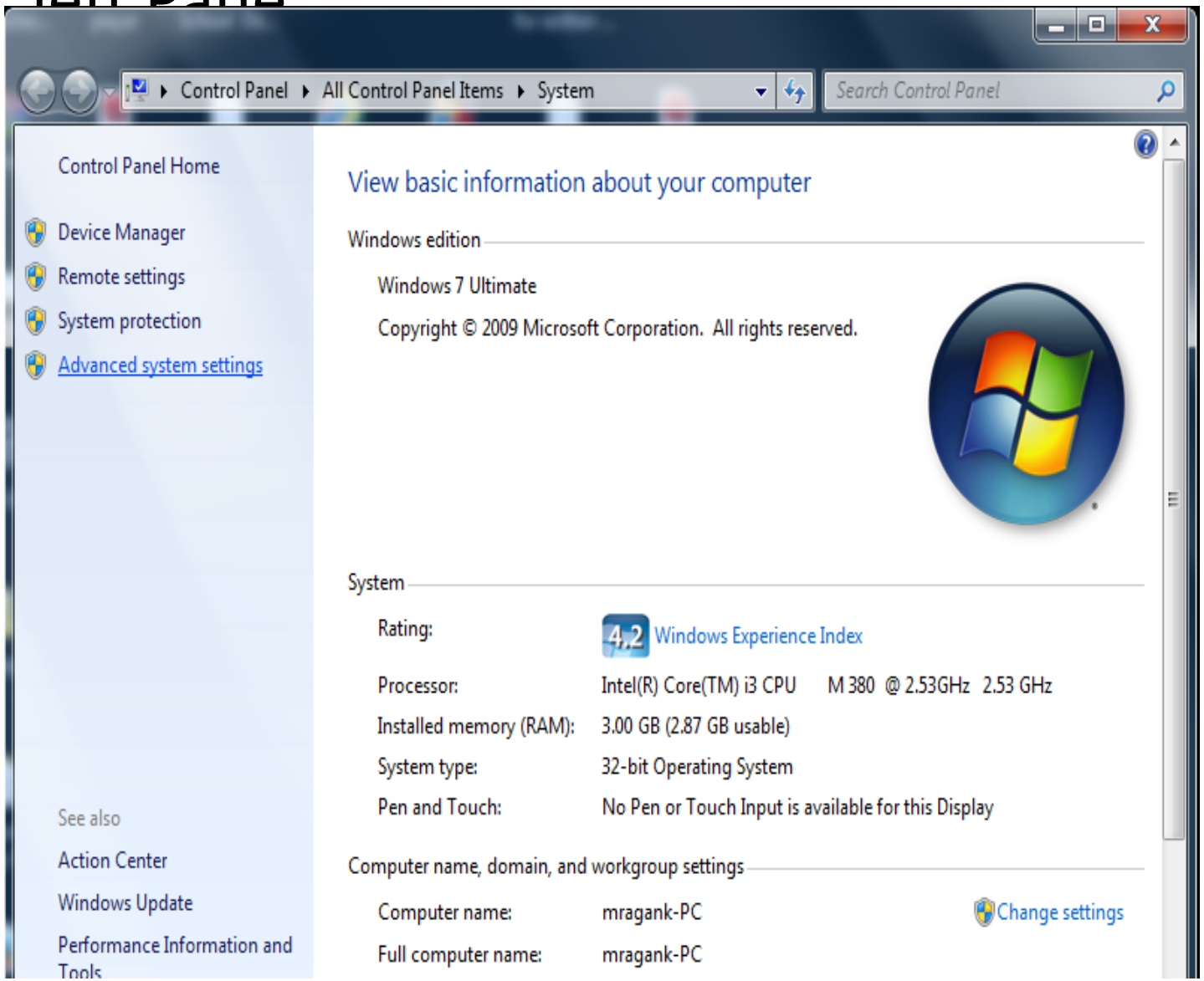
System

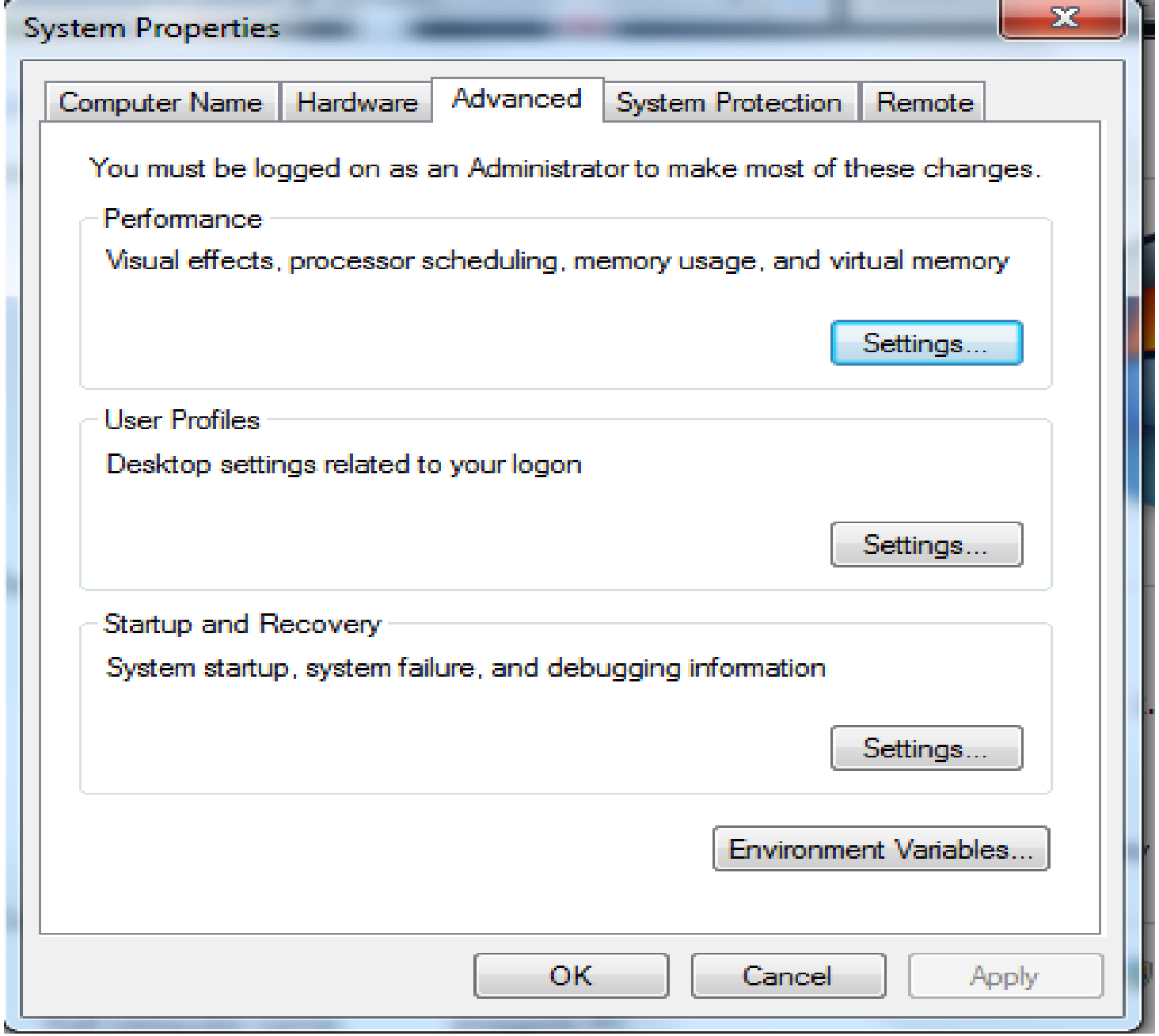
Rating:	4.2 Windows Experience Index
Processor:	Intel(R) Core(TM) i3 CPU M 380 @ 2.53GHz 2.53 GHz
Installed memory (RAM):	3.00 GB (2.87 GB usable)
System type:	32-bit Operating System
Pen and Touch:	No Pen or Touch Input is available for this Display

Computer name, domain, and workgroup settings

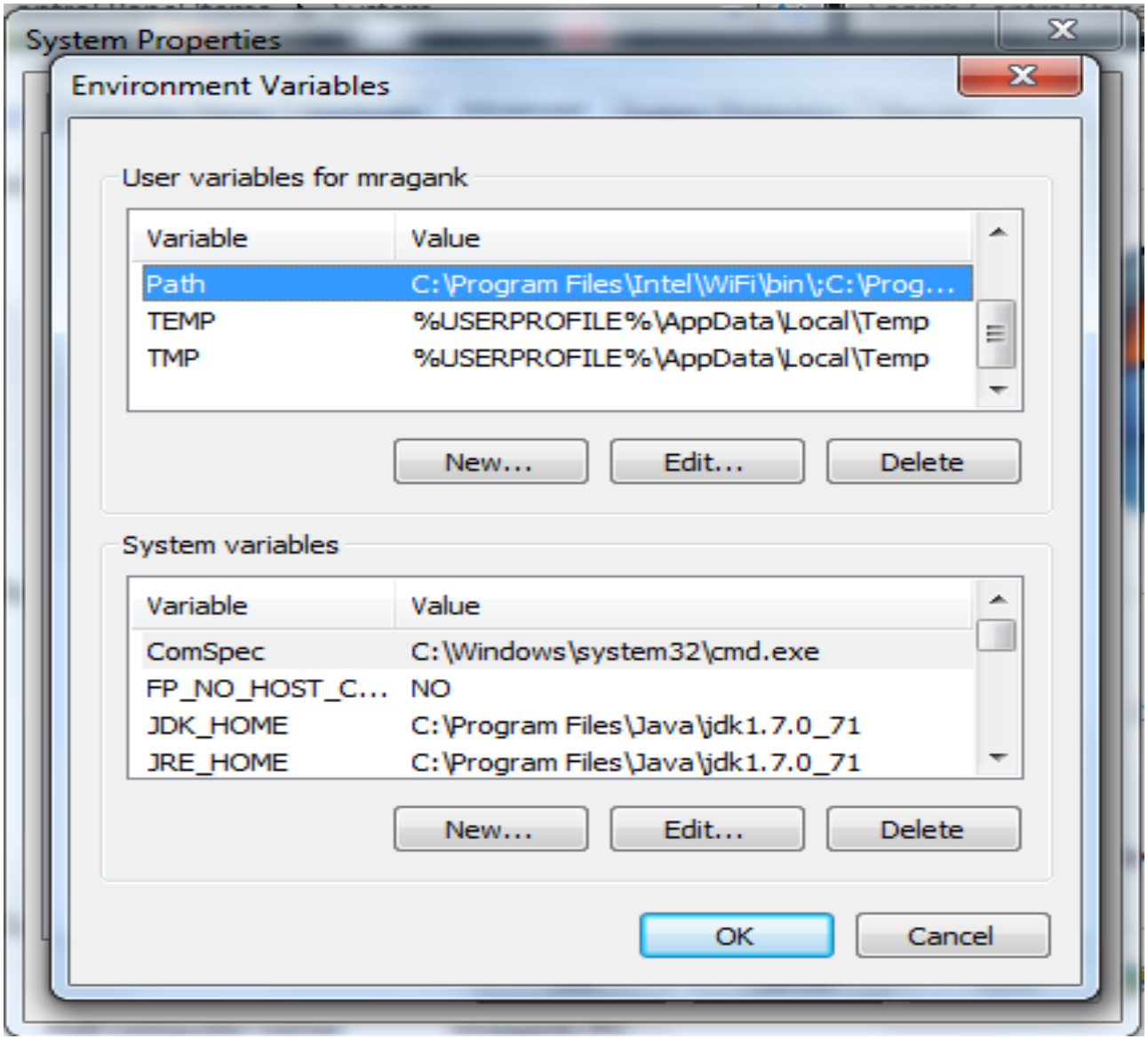
Computer name:	mragank-PC	Change settings
Full computer name:	mragank-PC	
Computer description:		

1. Now select “Advance System Settings” on left Pane

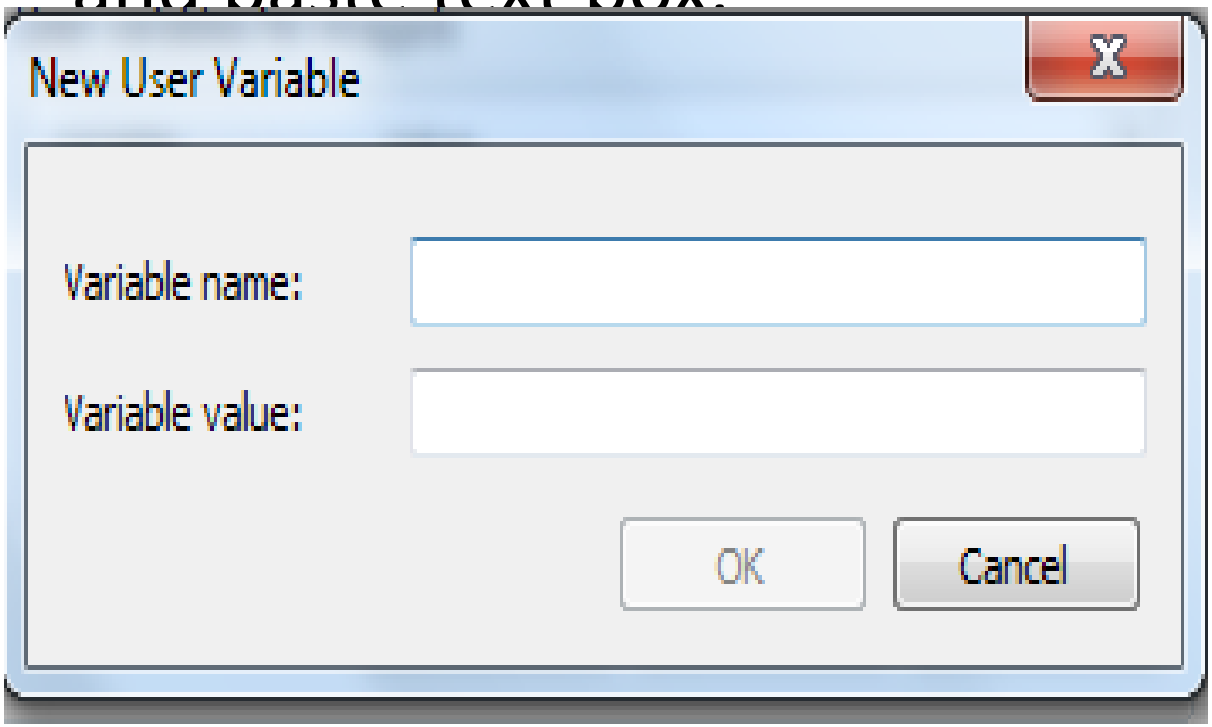



System Properties
Computer Name
Hardware
Advanced
System Protection
Remote
You must be logged on as an Administrator to make most of these changes.
Performance
Visual effects, processor scheduling, memory usage, and virtual memory
Settings...
User Profiles
Desktop settings related to your logon
Settings...
Startup and Recovery
System startup, system failure, and debugging information
Settings...
Environment Variables...
OK
Cancel
Apply

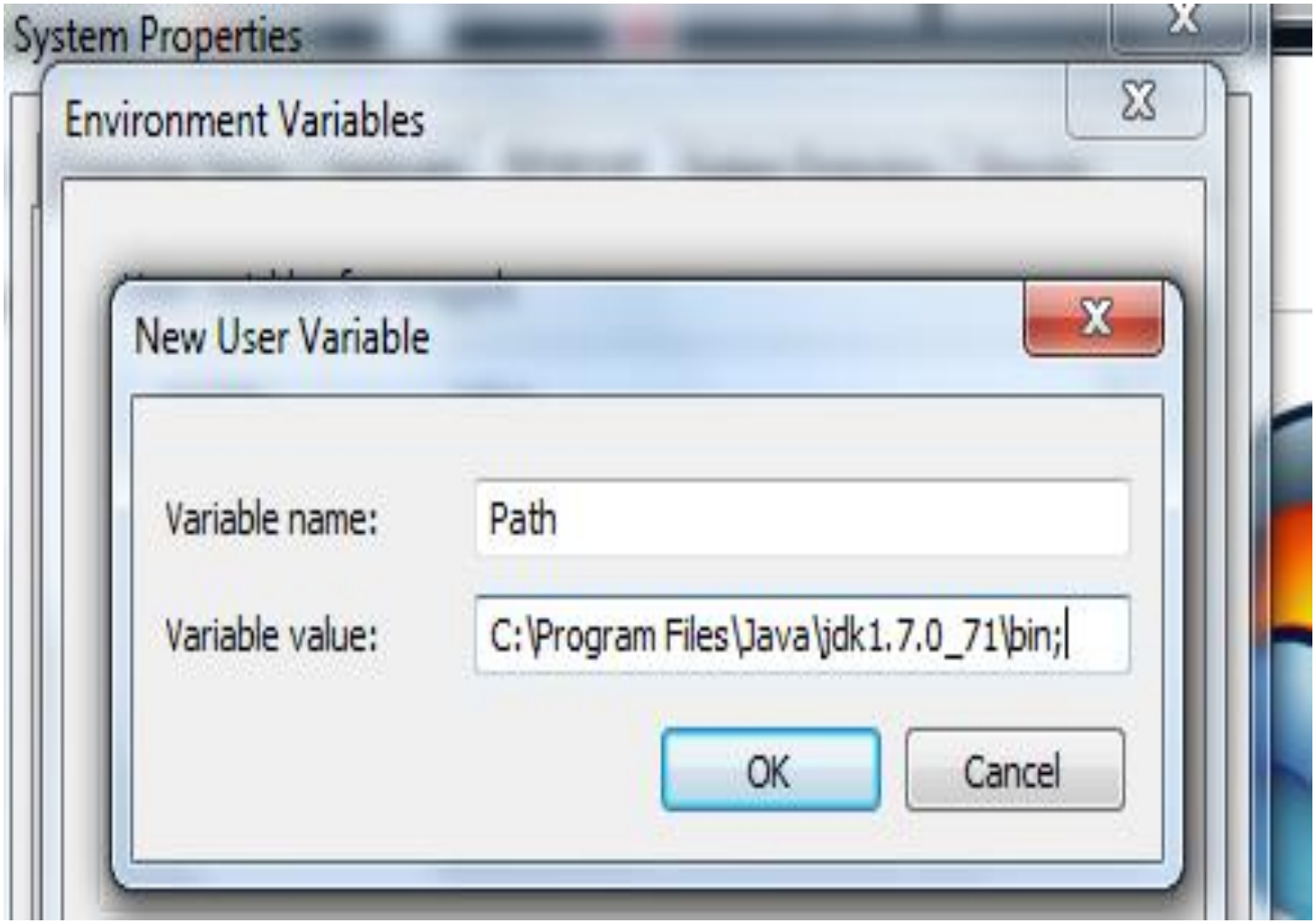
1. Click the Environment Variables button and then



1. Click on new Button and then provide a value “Variable name” and “Variable value”
2. Variable name=Path
3. Variable value=copy the path of java from bin(C:\Program Files(X86)\Java\Jdk1.7\bin) and paste text box.



1. Set the path



- Now type the javac command

C:\Windows\system32\cmd.exe

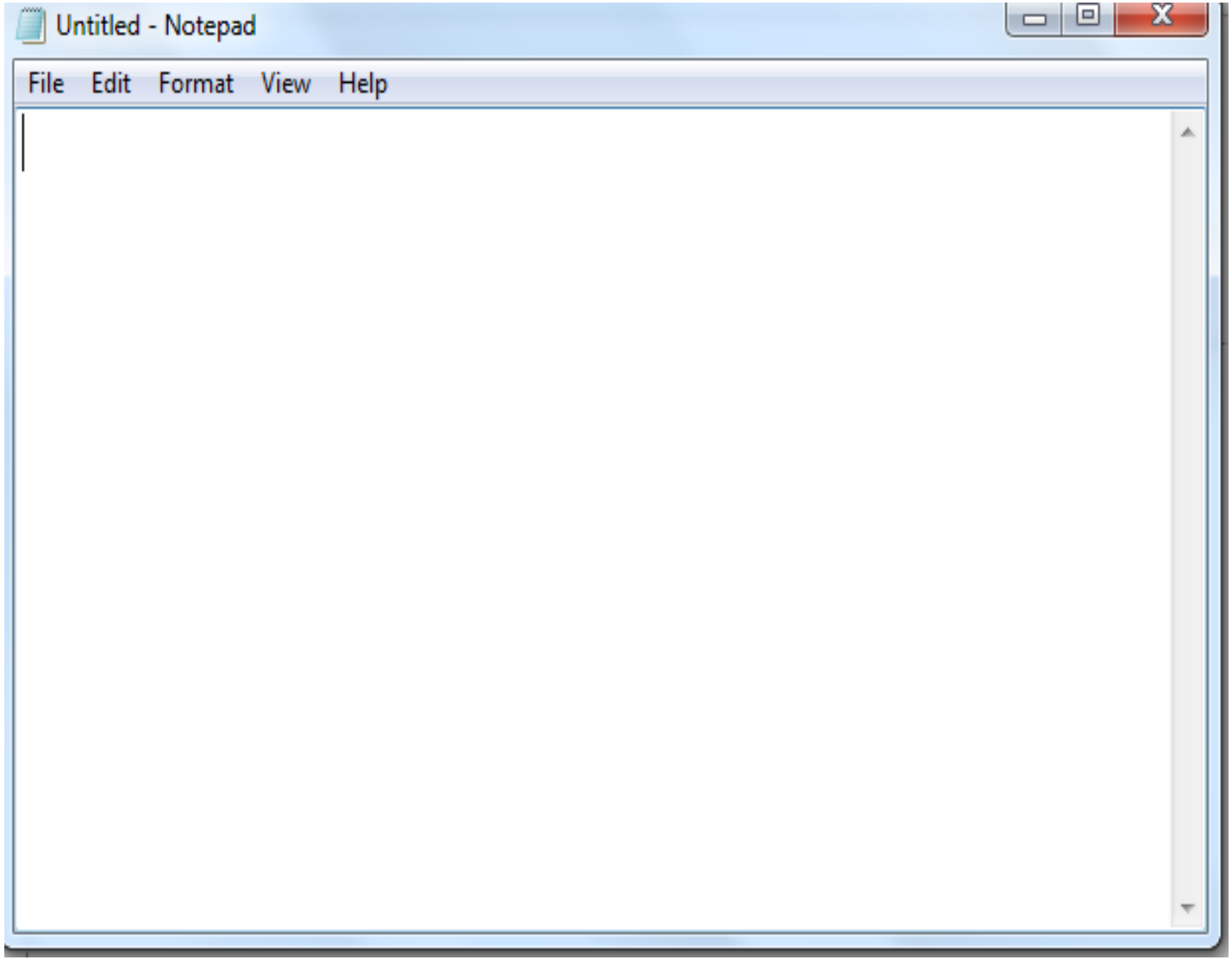
```
C:\Users\mragank>javac
Usage: javac <options> <source files>
where possible options include:
  -g                         Generate all debugging info
  -g:none                    Generate no debugging info
  -g:{lines,vars,source}     Generate only some debugging info
  -nowarn                    Generate no warnings
  -verbose                   Output messages about what the compiler is doing
  -deprecation               Output source locations where deprecated APIs are u
sed
  -classpath <path>          Specify where to find user class files and annotati
on processors
  -cp <path>                 Specify where to find user class files and annotati
on processors
  -sourcepath <path>         Specify where to find input source files
  -bootclasspath <path>      Override location of bootstrap class files
  -extdirs <dirs>            Override location of installed extensions
  -endorseddirs <dirs>       Override location of endorsed standards path
  -proc:{none,only}          Control whether annotation processing and/or compil
ation is done.
  -processor <class1>[,<class2>,<class3>...] Names of the annotation processors
to run; bypasses default discovery process
  -processorpath <path>      Specify where to find annotation processors
  -d <directory>             Specify where to place generated class files
  -s <directory>             Specify where to place generated source files
  -implicit:{none,class}     Specify whether or not to generate class files for
implicitly referenced files
  -encoding <encoding>       Specify character encoding used by source files
  -source <release>          Provide source compatibility with specified release

  -target <release>          Generate class files for specific VM version
  -version                   Version information
  -help                      Print a synopsis of standard options
  -Akey[=value]              Options to pass to annotation processors
  -X                         Print a synopsis of nonstandard options
  -J<flag>                   Pass <flag> directly to the runtime system
  -Werror                    Terminate compilation if warnings occur
  @<filename>                Read options and filenames from file


C:\Users\mragank>
```

First Example of Java

1. Open Notepad



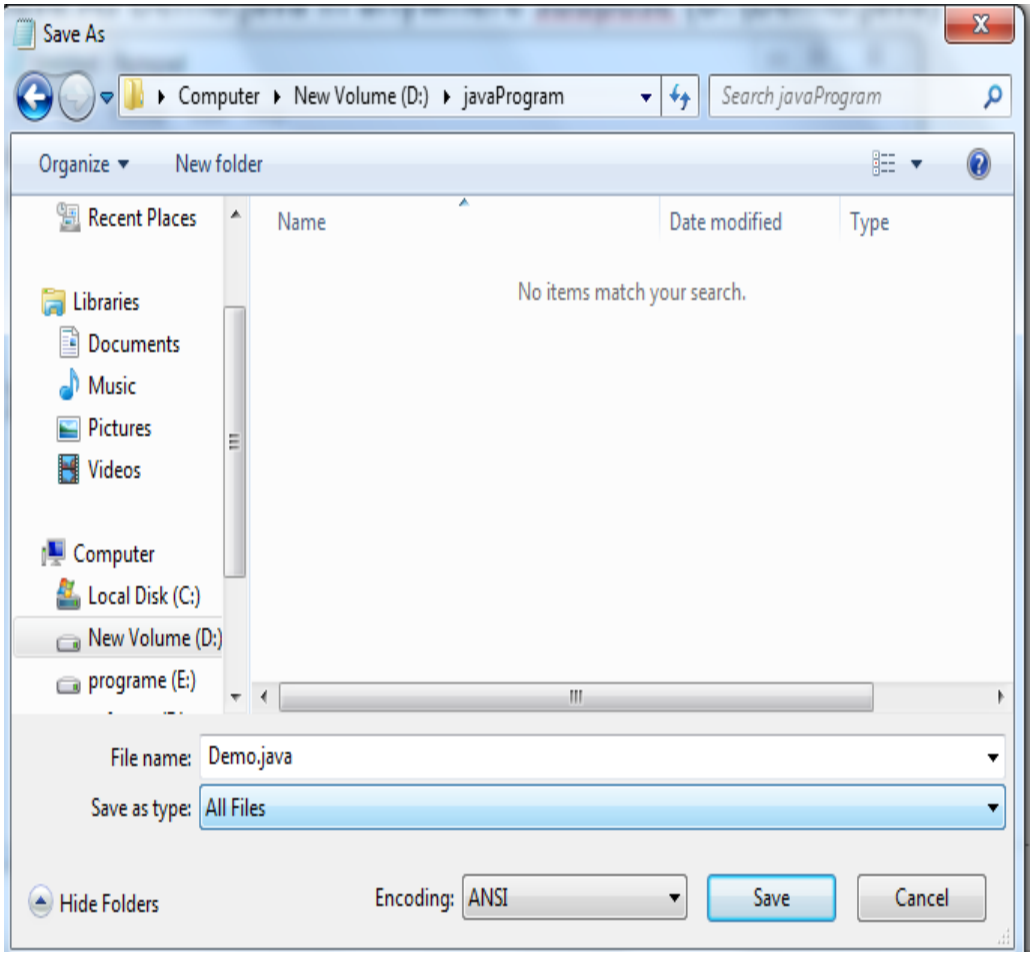
Type a simple hello Java Program

Untitled - Notepad

File Edit Format View Help

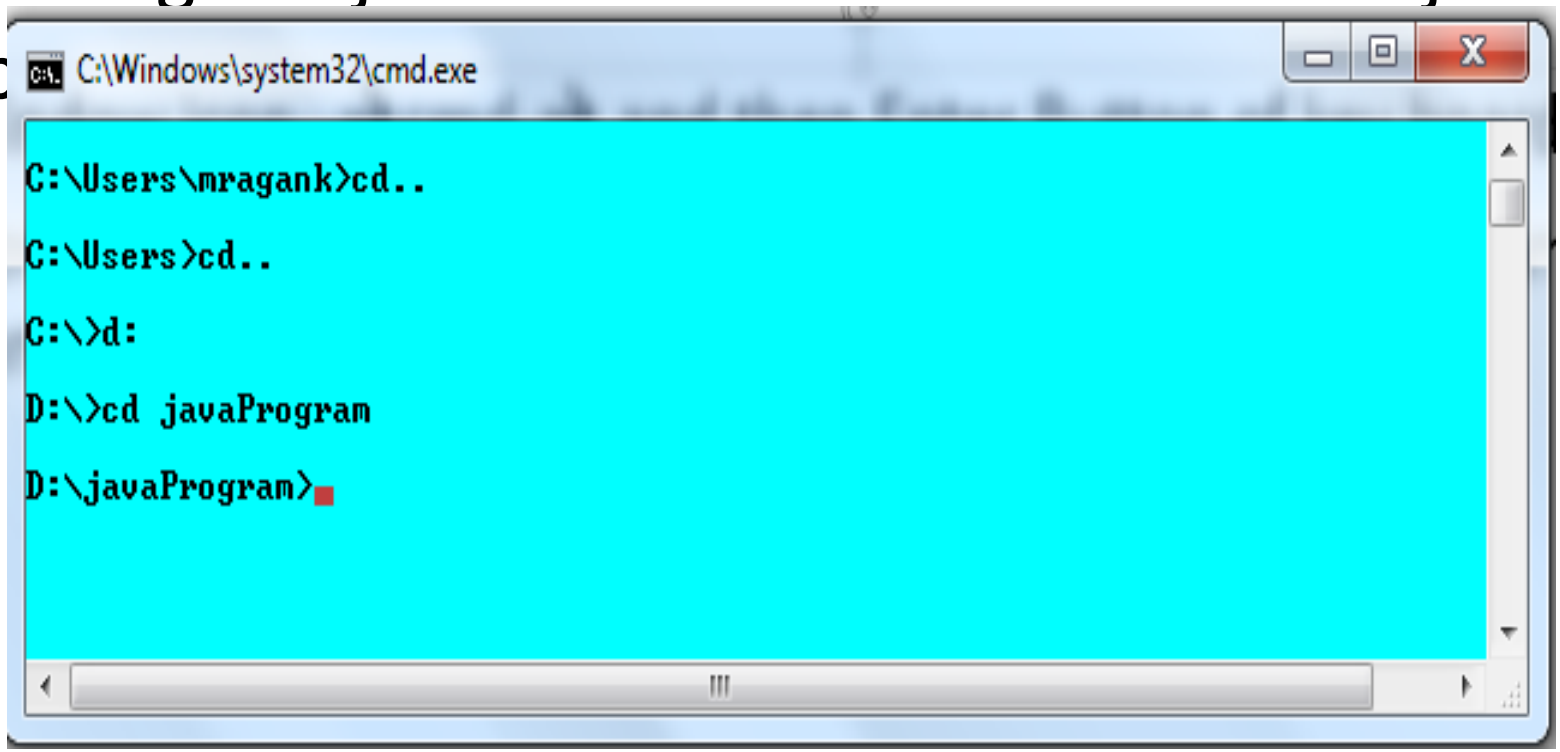
```
Class Demo
{
        public static void main(String args[])
        {
                System.out.println("Hello Java");
        }
}
```

1. Save As Demo.java in anywhere suppose (D:\javaProgram\Demo.java)

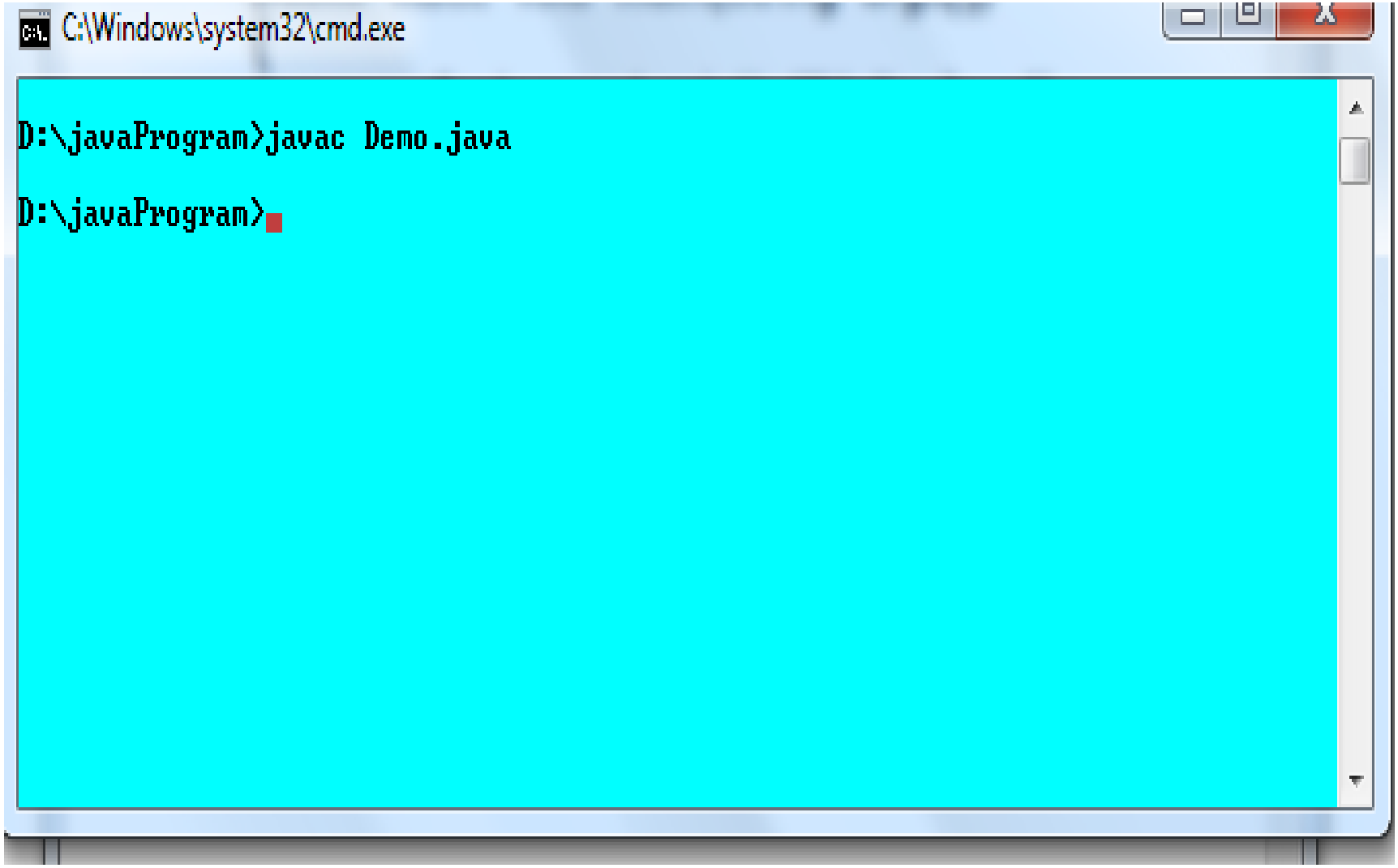


Open the Command Prompt

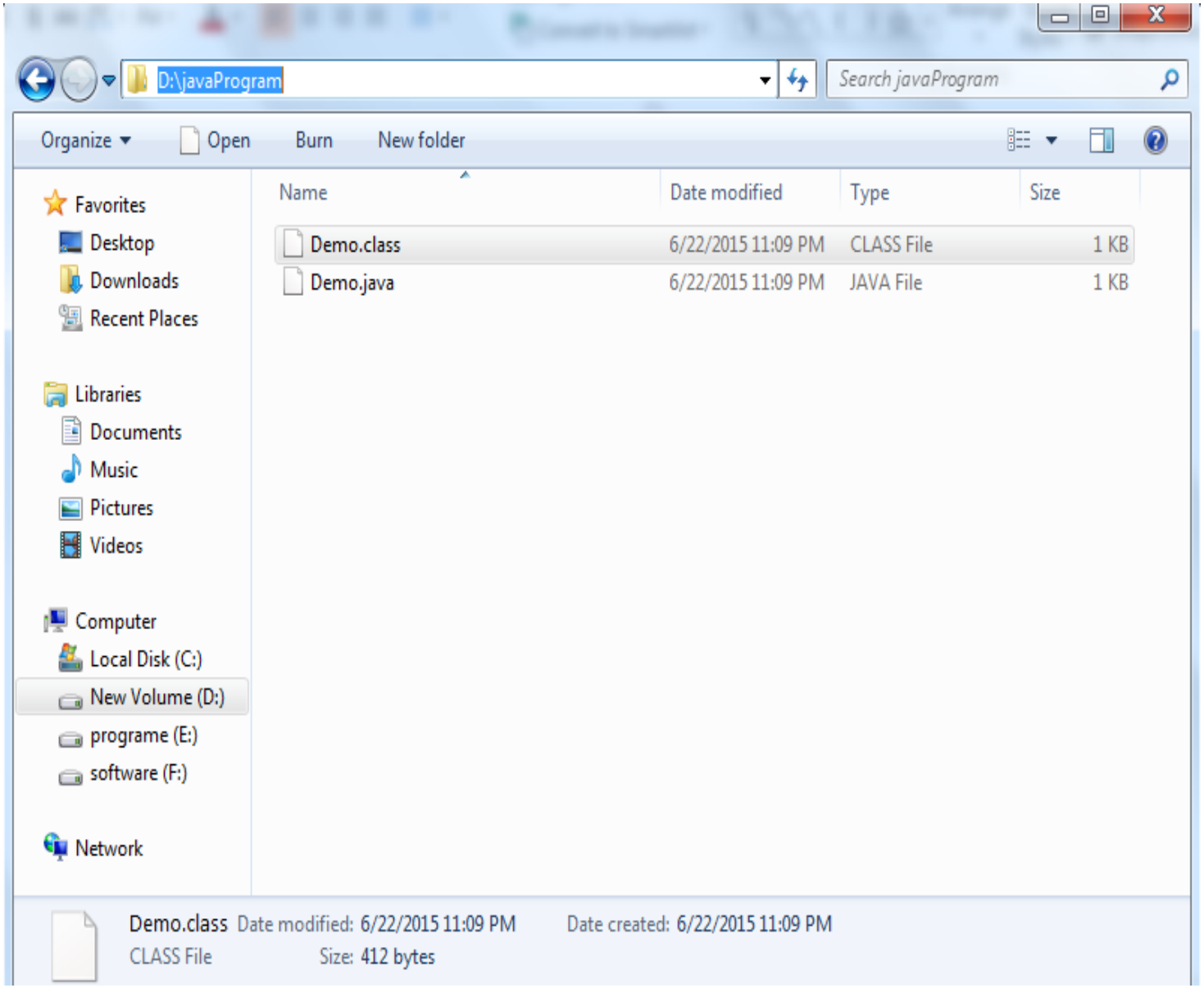
- Window icon--→cmd-→ and then Enter Button of key board
- Now go to java source file location where java p



- Now Type the Java Compiler command with "javac filename.java"



- Now check the .class file generated by



- Now run the java Program by the command of "java filename" and See the output

C:\Windows\system32\cmd.exe

```
D:\javaProgram>javac Demo.java

D:\javaProgram>java Demo
Hello Java

D:\javaProgram>
```

Bytecode…

- ***Byte-code*** **is a standardized machine independent, low level language. The byte code files are loaded and interpreted at the client's machine by a special program called Java Virtual Machine(JVM).**

- **For Example : An HTML document downloaded to your machine by a browser might embed a Java data entry applet. When we activate this applet, the byte code files are executed by the browser's JVM.**

Bytecode…

- **When a user runs a Java Program , it is upto the JVM to load, possibly verify and then execute it.**

- **The JVM can perform this function from within a browser or any other container program or directly on top of the operating system.**

What actually JVM does...

- **It validates the requested byte codes verifying that they pass various formatting and security checks. This is a security feature known as**

  Byte-code-verifier.

- **It allocates memory for the incoming Java class files and guarantees that the security of JVM is not violated. This is known as *class loader.***
- **It interprets the byte-code instructions found in the class files to execute the program.**

Java Development Kit…

- **The JDK comes with a collection of tools that are used for developing and running Java Programs.**

- **Appletviewer (for viewing Java applets)**
- **javac (Java compiler)**
- **java (Java Interpreter)**
- **javap (Java diassembler)**
- **javah (for C header files)**
- **jdb (Java debugger)**

First Java Application Program...

- ***Explanation:***

- **For most computer languages, the name of file that holds the source code to a program is arbitrary. However this is not the case with Java. The first thing you must learn about Java is that the name you give to a source file should match the name of the class holds the main() method. So, in our case name of program is none other than Firstapp.java. This is because the source file is officially called a compilation unit. Extension is also four character long, so your OS must support long extensions.**

First Java Application Program...

Explanation:

First statement import java.lang.*;

The purpose of this statement to instruct interpreter to load language package lang.

Second statement class Firstapp

Declares a class name firstapp so as to place everything inside this class.

Third statement public static void main(String args[ ])

Defines a method main. This is the starting point for

First Java Application Program...

- **the interpreter to begin the execution of program. Here *public* is an access specifier that declares the main method as unprotected and therefore making it accessible to all other classes.**

- **Next appears the keyword *static* which declares this method as one that belongs to the entire class and not a part of any objects of the class. The main must always be declared as static since the interpreter uses this method before any objects are created. The type modifier *void* states that main method does not return any value.**

First Java Application Program...

- In main(), there is only one parameter String args[ ]
- declared a parameter named args, which is an array of objects of the class String. Objects of type String store character strings. args receives any command-line arguments present when the program is executed. This program does not make use of this information.

First Java Application Program...

- **Fourth statement is**

System.out.println("This is First Application");

- **The *println* mehtod is a member of the *out* object, which is a static data member of *System* class. This line prints the string to the screen. The method println always appends a newline character to the end of the string. So for the output to be printed on the same line use print in place of println.**

First Java Application Program...

- To **compile** the Java program, execute the compiler, **javac**, specifying the name of the source file on command line:
- **C:\> javac Firstapp.java**
- **The javac compiler creates a file called Firstapp.class that contains the bytecode version of the program. The bytecode is the intermediate representation of your program that contains instructions the Java Interpreter will execute. So, to run your program you use Java interpreter java.**
- **C:\> java Firstapp**

First Java Application Program...

- **When Java source code is compiled each individual class is put its own output file named after the class and using the .class extension. This is why it is important to give source the same name as the class they contain, so when you execute your program you are actually executing the class by the interpreter. It will automatically search for a file by that name that has the .class extension. If it finds the file, it will execute the code contained in the specified class.**