Java Programming

Java Development Kit (JDK)

- The Java Development Kit (JDK) is a set of tools that developers use to develop Java applications.
- It includes the Java compiler, debugger, and other tools necessary for Java development.
- JDK is essential for compiling Java code into bytecode that can be executed by the Java Virtual Machine (JVM).

Java Development Kit (JDK)

Java Runtime Environment (JRE)

- The Java Runtime Environment (JRE) is a part of the JDK that enables Java programs to run on a computer.
- It includes the Java Virtual Machine (JVM) and libraries necessary for running Java applications.
- JRE is required on any machine where a Java program needs to be executed.

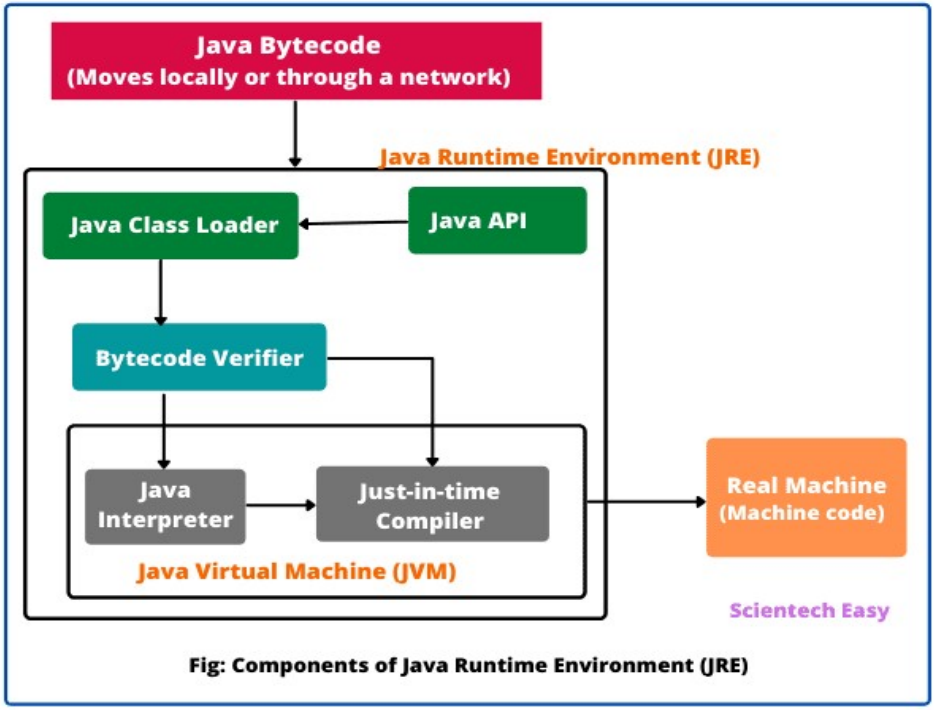
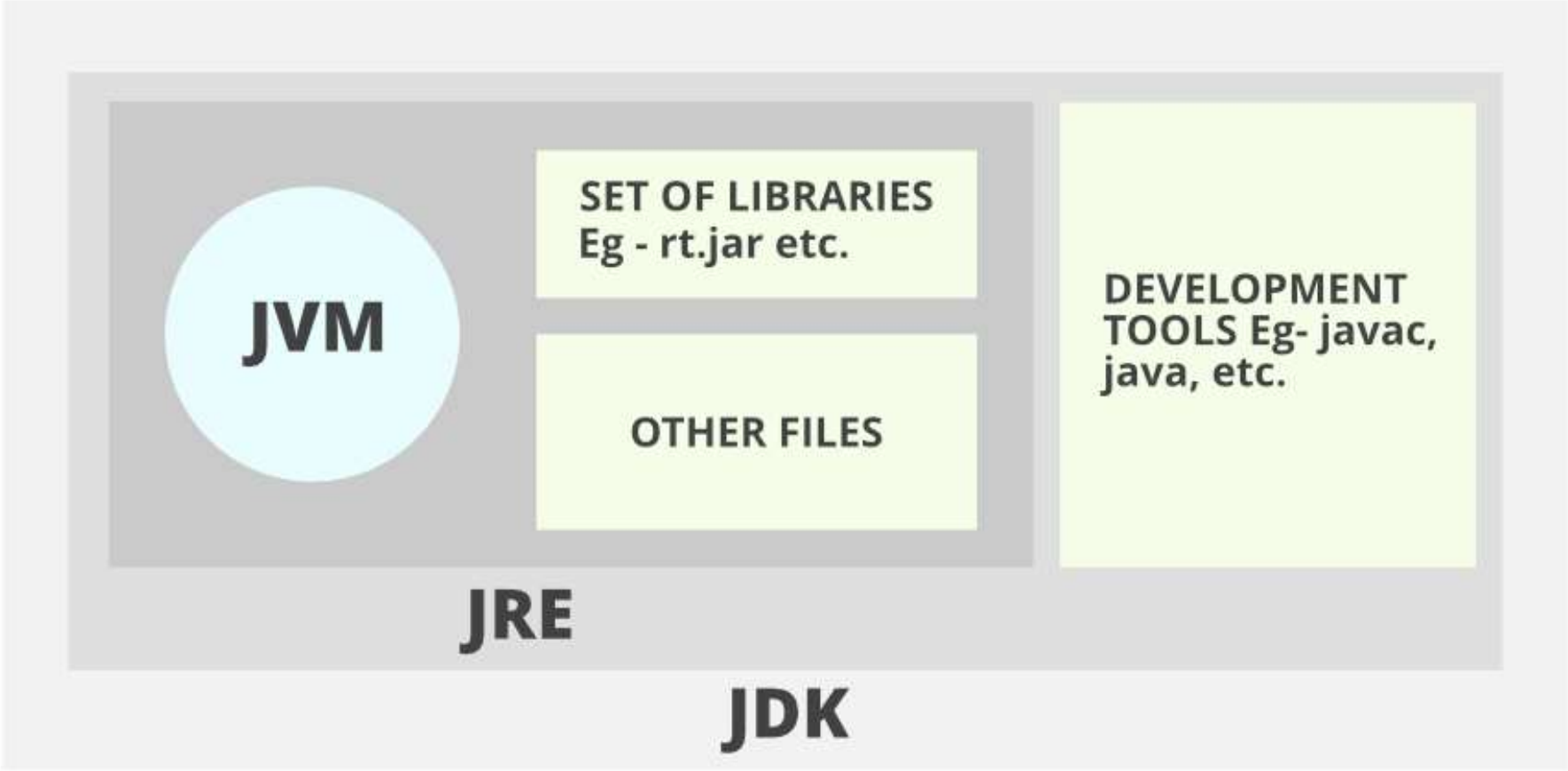


Fig: Components of Java Runtime Environment (JRE)



What is Java IDE?

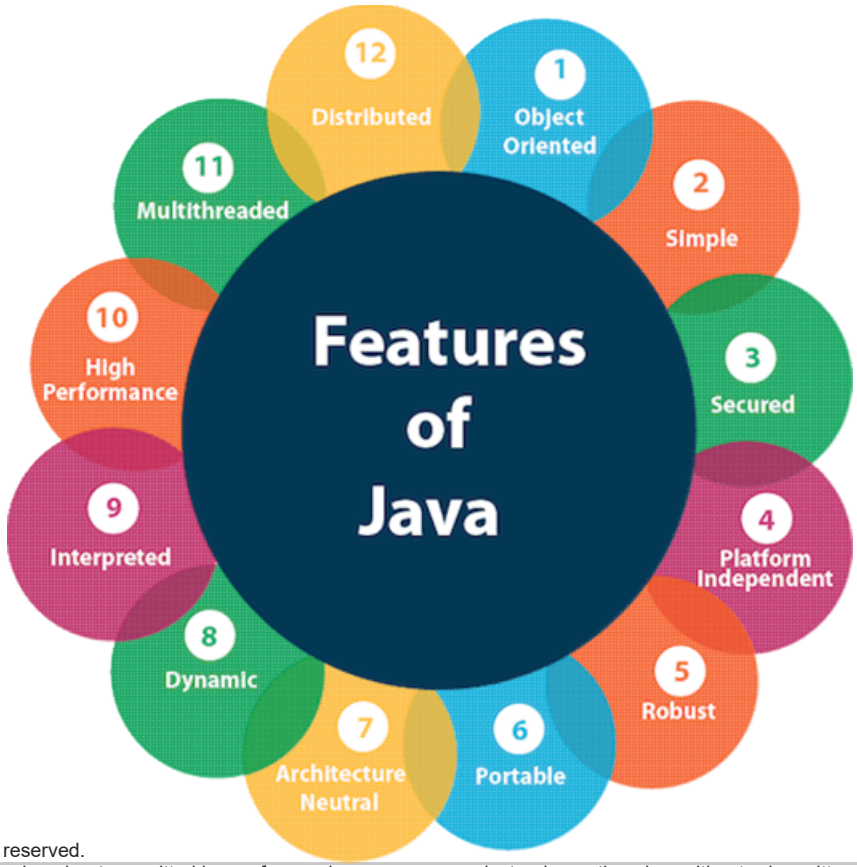
- Java IDE (Integrated Development Environment) is a software application that enables users to write and debug Java programs more easily.
- Most IDEs have features such as syntax highlighting and code completion that helps users to code more easily. Usually, Java IDEs include a code editor, a compiler, a debugger, and an interpreter that the developer may access via a single graphical user interface.
- Java IDEs also provide language-specific elements such as Maven, Ant building tools, Junit, and TestNG for testing.

The best Java IDEs in the world:

- Eclipse
- NetBeans
- IntelliJ IDEA
- BlueJ
- JCreator
- JDeveloper
- MyEclipse
- Greenfoot
- DrJava
- Xcode
- Codenvy

Features of Java

- The primary objective of Java programming language creation was to make it portable, simple and secure programming language



Java Installation Process

- Java –version

```
C:\Users\timbuchalka>java -version
java version "17.0.4.1" 2022-08-18 LTS
Java(TM) SE Runtime Environment (build 17.0.4.1+1-LTS-2)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.4.1+1-LTS-2, mixed mode, sharing)
```

- Jshell (entering jshell)
- Jshell› /help intro
- Jshell› /help

Introduction to JShell

- The Java Shell tool (JShell) is an interactive tool for learning the Java programming language and prototyping Java code. It was introduced in JDK 9. JShell is a Read-Evaluate-Print Loop tool (REPL), which evaluates declarations, statements, and expressions as they are entered and immediately shows the results. The tool is run from the command line.

Why Use JShell?

Using JShell, you can enter program elements one at a time, immediately see the result, and make adjustments as needed.

Java program development typically involves the following process:

- Write a complete program.
- Compile it and fix any errors.
- Run the program.
- Figure out what is wrong with it.
- Edit it.
- Repeat the process.

JShell helps you try out code and easily explore options as you develop your program. You can test individual statements, try out different variations of a method, and experiment with unfamiliar APIs within a JShell session. **JShell doesn't replace an IDE**. As you develop your program, paste code into JShell to try it out, and then paste working code from JShell into your program editor or IDE.

Starting and Stopping JShell

```
jshell> /list -all

  s1 : import java.io.*;
  s2 : import java.math.*;
  s3 : import java.net.*;
  s4 : import java.nio.file.*;
  s5 : import java.util.*;
  s6 : import java.util.concurrent.*;
  s7 : import java.util.function.*;
  s8 : import java.util.prefs.*;
  s9 : import java.util.regex.*;
 s10 : import java.util.stream.*;
   1 : 2 + 2
  e1 : list -all
  e2 : list -start

jshell> /list -start

  s1 : import java.io.*;
  s2 : import java.math.*;
  s3 : import java.net.*;
  s4 : import java.nio.file.*;
  s5 : import java.util.*;
  s6 : import java.util.concurrent.*;
  s7 : import java.util.function.*;
  s8 : import java.util.prefs.*;
  s9 : import java.util.regex.*;
 s10 : import java.util.stream.*;
```

JDK 9 or higher must be installed on your system. If your path doesn't include the bin directory, for example *java-home*`/jdk-16.0.1/bin`, then start the tool from within that directory.

The following example shows the command and the response from JShell. Text that you enter is shown in bold:

```
% jshell
|  Welcome to JShell -- Version 17.0.1
|  For an introduction type: /help intro

jshell>
```

Copy

The examples in this tutorial use the verbose mode. Verbose mode is recommended as you work through this tutorial so that what you see matches the examples. When you are more familiar with the tool, you might prefer to run in normal or a more concise mode.

To start JShell in verbose mode, use the `-v` option:

```
% jshell -v
```

Copy

To exit JShell, enter `/exit`:

```
jshell> /exit
|  Goodbye
```

Copy

Snippets

- JShell accepts Java statements; variable, method, and class definitions; imports; and expressions. These pieces of Java code are referred to as snippets.

Enter the following sample statement at the prompt, and review the output that is shown:

```
jshell> int x = 45
x ==> 45
|  created variable x : int
```

Copy

```
jshell> 2 + 2
$3 ==> 4
|  created scratch variable $3 : int

jshell> String twice(String s) {
   ...>     return s + s;
   ...> }
|  created method twice(String)

jshell> twice("Ocean")
$5 ==> "OceanOcean"
|  created scratch variable $5 : String
```

Copy

THANK YOU