ICTACADEMY

Module – I

Introduction to Java

Java™

What you will Learn

- Introduction to Java
- History of Java
- Features
- Applications
- JDK,JVM & JRE
- Comments, Variables, Data-Types & Type Casting

Introduction to Java

- Java is a **programming language and platform.**
- Java is a high level, robust, object-oriented and secure programming language.
- Java was developed by *Sun Microsystems* (which is now the subsidiary of Oracle) in the year 1995.

What is Platform?

- Any hardware or software environment in which a program runs, is known as a platform.
- Since Java has a runtime environment (JRE) and API, it is called a platform.
- Platform represents OS

History of Java

- The history of Java starts with the Green Team.

- Java team members (also known as **Green Team**), initiated this project to develop a language for digital devices such as set-top boxes, televisions, etc.

- However, it was suited for internet programming. Later, Java technology was incorporated by Netscape.

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

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

History of Java

ICTACADEMY

Why Java named "Oak"?

- **Why Oak?** Oak is a symbol of strength and chosen as a national tree of many countries like the U.S.A., France, Germany, Romania, etc.

- In 1995, Oak was renamed as **"Java"** because it was already a trademark by Oak Technologies.

Simple Example to understand Java

```
class Easy{
    public static void main(String args[]){
     System.out.println(“Hey,it’s Java");
    }
}
```

Why we learn Java

- Java is Easy to Learn
- Java has multiple Open Source Libraries
- Java has an abundant API
- Java has Powerful Development Tools
- Java is Free of Cost
- Java is Platform Independent

Where Java is used?

- Desktop Applications such as acrobat reader, media player, antivirus, etc
- Web Applications such as irctc.co.in, etc
- Enterprise Applications such as banking applications
- Mobile
- Games, Animations
- Smart Card
- Robotics

Features of Java

There are many features of java:

- Simple
- Object-Oriented
- Platform independent
- Secured
- Robust
- Portable
- High Performance
- Distributed

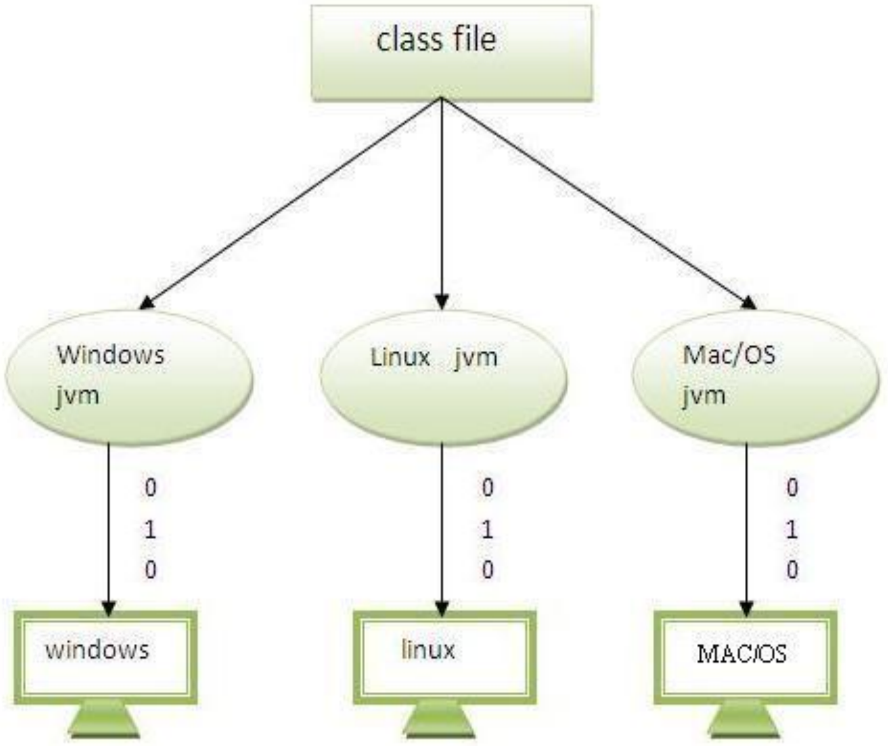
How Java is Platform Independent

- Java is a platform independent programming language, because when you install jdk software on your system then automatically JVM are installed on your system.

- For every operating system separate JVM is available which is capable to read the .class file or byte code.

- When we compile your Java code then .class file is generated by javac compiler these codes are readable by JVM and every operating system have its own JVM so JVM is platform dependent but due to JVM java language is become platform independent.

- Java code can be run on multiple platforms e.g.Windows,Linux,SunSolaris,Mac/OS etc.
- Java code is compiled by the compiler and converted into bytecode.
- Thisbytecode is a platform independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere(WORA).



Difference between JDK,JVM & JRE

ICTACADEMY

JVM

- JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed.
- JVMs are available for many hardware and software platforms. JVM, JRE and JDK are platform dependent because configuration of each OS differs. But, Java is platform independent.

Difference between JDK,JVM & JRE

The JVM performs following main tasks:

- Loads code
- Verifies code
- Executes code
- Provides runtime environment

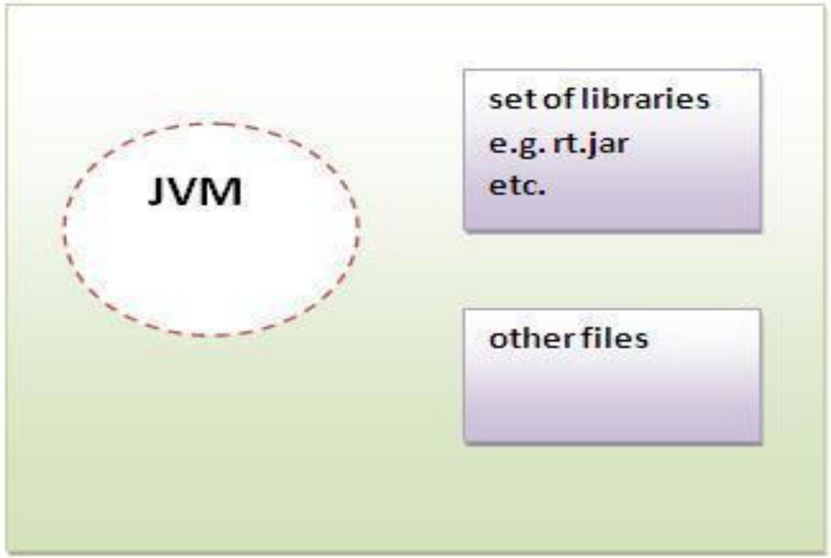
What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.

JRE

- JRE is an acronym for Java Runtime Environment.It is used to provide runtime environment.It is the implementation of JVM.

- It physically exists.It contains set of libraries + other files that JVM uses at runtime.

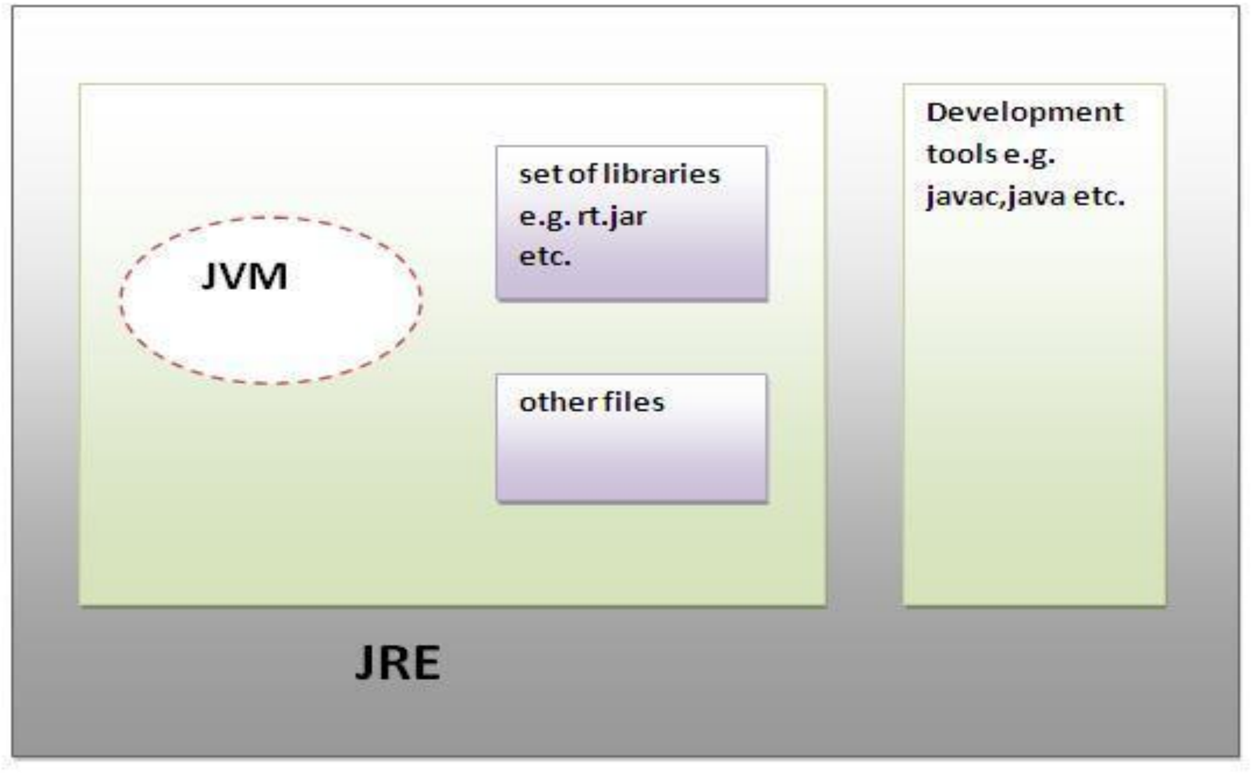


JRE

JDK

- The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets.

- It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development.


set of libraries e.g. rt.jar etc.
Development tools e.g. javac,java etc.
JVM
other files
JRE

JDK

Variable

A variable is a container which holds the value while the Java program is executed.

Variable is a name of memory location.

Variable Types

There are three types of variables in Java:

- local variable
- instance variable
- static variable

Local, Instance & Static variable

Local Variable

- A variable declared inside the body of the method is called local variable.

- A local variable cannot be defined with "static" keyword.

Local, Instance & Static variable

ICTACADEMY

Instance Variable

- A variable declared inside the class but outside the body of the method, is called instance variable.
- It is not declared as static.

Local, Instance & Static variable

ICTACADEMY

Static Variable

- A variable which is declared as static is called static variable. It cannot be local.

- You can create a single copy of static variable and share among all the instances of the class.

Example to Understand Variables

```
class variable{
int data=50;//instance variable
static int var=100;//static variable
void method(){
int var2=90;//local variable
}
}//end of class
```

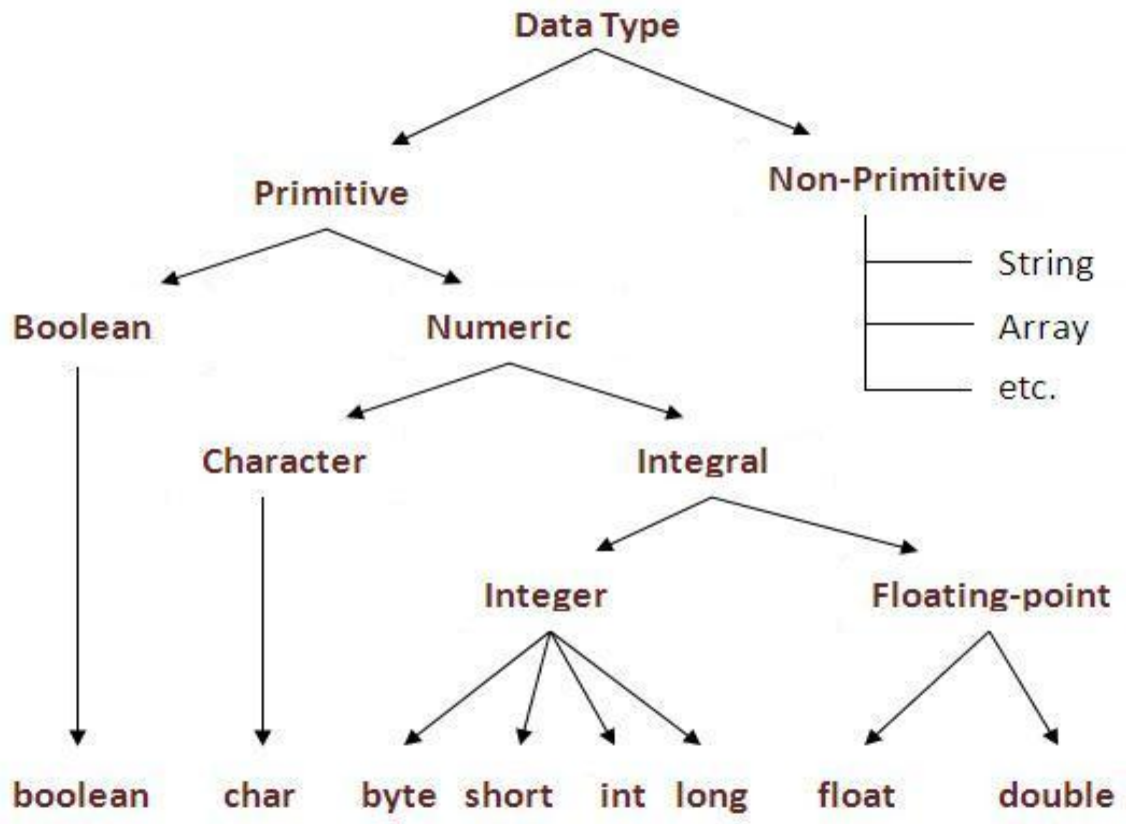
Data-Types

Data types specify the different sizes and values that can be stored in the variable. There are two types of data types in Java:

- **Primitive data types:** The primitive data types include boolean, char, byte, short, int, long, float and double.

- **Non-primitive data types:** The non-primitive data types include Classes, Interfaces, and Arrays.

ICTACADEMY


Data Type
Primitive
Non-Primitive
String
Array
etc.
Boolean
Numeric
Character
Integral
Integer
Floating-point
boolean
char
byte
short
int
long
float
double

Data Type	Default Value	Default size
boolean	false	1 bit
char	'\u0000'	2 byte
byte	0	1 byte
short	0	2 byte
int	0	4 byte
long	0L	8 byte
float	0.0f	4 byte
double	0.0d	8 byte

Type Casting

- **type casting** is a method or process that converts a data type into another data type in both ways manually and automatically.

- The automatic conversion is done by the compiler and manual conversion performed by the programmer.

Type Casting

- **Widening Casting** (automatic) – converting a smaller type to a larger type size
  byte -> short -> char -> int -> long -> float -> double
-
  Narrowing Casting (manual) – converting a larger type to a smaller size type
  double -> float -> long -> int -> char -> short -> byte

Type Casting

ICTACADEMY

- **Widening Casting(Example)**

```
public class Main {
 public static void main(String[] args) {
   int myInt = 9;
   double myDouble = myInt;
   System.out.println(myInt);
   System.out.println(myDouble);
 }
}
```

Type Casting

ICTACADEMY

- **Widening Casting(Example)**

```
public class Main {
 public static void main(String[] args) {
   double myDouble = 9.78;
   int myInt = (int) myDouble;
   System.out.println(myDouble);
   System.out.println(myInt);
 }
}
```

Java Comments

The Java comments are the statements that are not executed by the compiler and interpreter.

The comments can be used to provide information or explanation about the variable, method, class or any statement.

It can also be used to hide program code.

Java Comments – Types

There are three types of comments in Java.

- Single Line Comment – comment only one line
- Multi Line Comment – comment multiple lines of code.
- Documentation Comment – create documentation API.

Java Comments – Example

Single Line Comment

```
public class Comment1 {
public static void main(String
[] args) {
    int a=10;//Here, a is a varia
ble
    System.out.println(a);
}
}
```

MultiLine Comment

```
public class Comment1 {
public static void main(String
[] args) {

/* Let's declare and
 print variable in java. */

    int a=10;//Here, a is a varia
ble
    System.out.println(a);
}
}
```

Java Comments – Example

Documentation Support

```
/** The Calculator class provides methods
to get addition and subtraction of given 2 n
umbers.*/
public class Calculator {
/** The add() method returns addition of gi
ven numbers.*/
public static int add(int a, int b){return a+b;
}
/** The sub() method returns subtraction o
f given numbers.*/
public static int sub(int a, int b){return a-b;
}
}
```