

**Features of Java are as follows:**

**1. Compiled and Interpreted**

Basically a computer language is either compiled or interpreted. Java comes together both

these approach thus making Java a two-stage system.

Java compiler translates Java code to Bytecode instructions and Java Interpreter generate

machine code that can be directly executed by machine that is running the Java program.

**2. Platform Independent and portable**

Java supports the feature portability. Java programs can be easily moved from one computer

system to another and anywhere. Changes and upgrades in operating systems, processors and

system resources will not force any alteration in Java programs. This is reason why Java has

become a trendy language for programming on Internet which interconnects different kind of

systems worldwide. Java certifies portability in two ways.

First way is, Java compiler generates the bytecode and that can be executed on any machine.

Second way is, size of primitive data types are machine independent.

**3. Object- oriented**

Java is truly object-oriented language. In Java, almost everything is an Object. All program code

and data exist in objects and classes. Java comes with an extensive set of classes; organize in

packages that can be used in program by Inheritance. The object model in Java is trouble-free

and easy to enlarge.

**4. Robust and secure**

Java is a most strong language which provides many securities to make certain reliable code. It is

design as garbage –collected language, which helps the programmers virtually from all memory

management problems. Java also includes the concept of exception handling, which detain

serious errors and reduces all kind of threat of crashing the system.

Security is an important feature of Java and this is the strong reason that programmer use

this language for programming on Internet.

The absence of pointers in Java ensures that programs cannot get right of entry to memory

location without proper approval.

**5. Distributed**

Java is called as Distributed language for construct applications on networks which can

contribute both data and programs. Java applications can open and access remote objects on

Internet easily. That means multiple programmers at multiple remote locations to work together

on single task.

**6. Simple and small**

Java is very small and simple language. Java does not use pointer and header files, goto

statements, etc. It eliminates operator overloading and multiple inheritance.

**7. Multithreaded and Interactive**

Multithreaded means managing multiple tasks simultaneously. Java maintains multithreaded

programs. That means we need not wait for the application to complete one task before starting

next task. This feature is helpful for graphic applications.

8. High performance

Java performance is very extraordinary for an interpreted language, majorly due to the use of

intermediate bytecode. Java architecture is also designed to reduce overheads during runtime.

The incorporation of multithreading improves the execution speed of program.

**9. Dynamic and Extensible**

Java is also dynamic language. Java is capable of dynamically linking in new class, libraries,

methods and objects. Java can also establish the type of class through the query building it

possible to either dynamically link or abort the program, depending on the reply.

Java program is support functions written in other language such as C and C++, known as native

methods.Java supports the feature portability. Java programs can be easily moved from one computer

system to another and anywhere. Changes and upgrades in operating systems, processors and

system resources will not force any alteration in Java programs. This is reason why Java has

become a trendy language for programming on Internet which interconnects different kind of

systems worldwide. Java certifies portability in two ways.

First way is, Java compiler generates the bytecode and that can be executed on any machine.

Second way is, size of primitive data types are machine independent.

**3. Object- oriented**

Java is truly object-oriented language. In Java, almost everything is an Object. All program code

and data exist in objects and classes. Java comes with an extensive set of classes; organize in

packages that can be used in program by Inheritance. The object model in Java is trouble-free

and easy to enlarge.

**4. Robust and secure**

Java is a most strong language which provides many securities to make certain reliable code. It is

design as garbage –collected language, which helps the programmers virtually from all memory

management problems. Java also includes the concept of exception handling, which detain

serious errors and reduces all kind of threat of crashing the system.

Security is an important feature of Java and this is the strong reason that programmer use

this language for programming on Internet.

The absence of pointers in Java ensures that programs cannot get right of entry to memory

location without proper approval.

**5. Distributed**

Java is called as Distributed language for construct applications on networks which can

contribute both data and programs. Java applications can open and access remote objects on

Internet easily. That means multiple programmers at multiple remote locations to work together

on single task.

**6. Simple and small**

Java is very small and simple language. Java does not use pointer and header files, goto

statements, etc. It eliminates operator overloading and multiple inheritance.

**7. Multithreaded and Interactive**

Multithreaded means managing multiple tasks simultaneously. Java maintains multithreaded

programs. That means we need not wait for the application to complete one task before starting

next task. This feature is helpful for graphic applications.

8. High performance

Java performance is very extraordinary for an interpreted language, majorly due to the use of

intermediate bytecode. Java architecture is also designed to reduce overheads during runtime.

The incorporation of multithreading improves the execution speed of program.

**9. Dynamic and Extensible**

Java is also dynamic language. Java is capable of dynamically linking in new class, libraries,

methods and objects. Java can also establish the type of class through the query building it

possible to either dynamically link or abort the program, depending on the reply.

Java program is support functions written in other language such as C and C++, known as native

methods.

Java Virtual machine: As we know that all programming language compilers convert the source code to machine

code.Same job done by Java Compiler to run a Java program, but the difference is that Java

compiler convert the source code into Intermediate code is called as bytecode. This machine

is called the Java Virtual machine and it exits only inside the computer memory.

Following figure shows the process of compilation.

**Source Code Byte Code**

The Virtual machine code is not machine specific. The machine specific code is generated. By Java

interpreter by acting as an intermediary between the virtual machine and real machines shown below

**Virtual machine Real Machine**

Java Object Framework act as the intermediary between the user programs and the virtual machine



which in turn act as the intermediary between the operating system and the Java Object Framework. Java Environment:

Java environment includes a number of development tools, classes and methods. The

development tools are part of the system known as Java Development Kit (JDK) and the classes

and methods are part of the Java Standard Library (JSL), also known as the Application

Programming Interface (API).

Java Development kit (JDK) – The JDK comes with a set of tools that are used for developing

and running Java program. It includes:

1. Appletviewer( It is used for viewing the applet)

2. Javac(It is a Java Compiler)

3. Java(It is a java interpreter)

4. Javap(Java diassembler,which convert byte code into program

description)

5. Javah(It is for java C header files)

6. Javadoc(It is for creating HTML document)

7. Jdb(It is Java debugger)

For compiling and running the program we have to use following commands:

a) javac (Java compiler)

In java, we can use any text editor for writing program and then save that program with

―.java‖ extension. Java compiler convert the source code or program in bytecode and

interpreter convert ―.java‖ file in ―.class‖ file.

Syntax:

C:\javac filename.java

If my filename is ―abc.java‖ then the syntax will be

C:\javac abc.java

b) java(Java Interpreter)

As we learn that, we can use any text editor for writing program and then save that

program with ―.java‖ extension. Java compiler convert the source code or program in

bytecode and interpreter convert ―.java‖ file in ―.class‖ file.

Syntax:

C:\java filename

If my filename is abc.java then the syntax will be

C:\java abc

1.5 Simple Java Program:

class FirstProgram

{

public static void main(String args[])

{

System.out.println(―This is my first program‖);

}

}

The file must be named ―FirstProgram.java‖ to equivalent the class name containing the

main method.

Java is case sensitive. This program defines a class called ―FirstProgram‖.

A class is an object oriented term. It is designed to perform a specific task. A Java class is

defined by its class name, an open curly brace, a list of methods and fields, and a close

curly brace.

The name of the class is made of alphabetical characters and digits without spaces, the

first character must be alphabetical.

The line ―public static void main (String [] args )‖ shows where the program will start

running. The word main means that this is the main method –

The JVM starts running any program by executing this method first.

The main method in ―FirstProgram.java‖ consists of a single statement

System.out.println("This is my first program");

The statement outputs the character between quotes to the console.

Above explanation is about how to write program and now we have to learn where to

write program and how to compile and run the program.

For this reason, the next explanation is showing the steps.

1. Edit the program by the use of Notepad.

2. Save the program to the hard disk.

3. Compile the program with the javac command.(Java compiler)

4. If there are syntax errors, go back to Notepad and edit the program.

5. Run the program with the java command.(Java Interpreter)

6. If it does not run correctly, go back to Notepad and edit the program.

7. When it shows result then stop

Data types:

A data type is a scheme for representing values. An example is int which is the Integer, a

data type. Values are not just numbers, but any manner of data that a computer can process.

The data type defines the kind of data that is represented by a variable.

As with the keyword class, Java data types are case sensitive.

There are two types of data types

primitive data type

non-pimitive data type

In primitive data types, there are two categories

numeric means Integer, Floating points

Non-numeric means Character and Boolean

In non-pimitive types, there are three categories

classes

arrays

interface

Following table shows the datatypes with their size and ranges.

Fig: Datatypes with size and range

2.1.1 Integer data type:

Integer datatype can hold the numbers (the number can be positive number or negative

number). In Java, there are four types of integer as follows:

byte

short

int

long

We can make ineger long by adding ‗l‘ or ‗L‘ at the end of the number.

2.1.2 Floating point data type:

It is also called as Real number and when we require accuracy then we can use it.

There are two types of floating point data type.

Data type Size (byte) Range

byte 1 -128 to 127

boolean 1 True or false

char 2 A-Z,a-z,0-9,etc.

short 2 -32768 to 32767

Int 4 (about) -2 million to 2 million

long 8 (about) -10E18 to 10E18

float 4 -3.4E38 to 3.4E18

double 8 -1.7E308 to 1.7E308 float

double

It is represent single and double precision numbers. The float type is used for single precision

and it uses 4 bytes for storage space. It is very useful when we require accuracy with small

degree of precision. But in double type, it is used for double precision and uses 8 bytes of starage

space. It is useful for large degree of precision.

2.1.3 Character data type:

It is used to store single character in memory. It uses 2 bytes storage space.

2.1.4 Boolean data type:

It is used when we want to test a particular condition during the excution of the program. There

are only two values that a boolean type can hold: true and false.

Boolean type is denoted by the keyword boolean and uses only one bit of storage. Variables:

Variables are labels that express a particular position in memory and connect

it with a data type.

The first way to declare a variable: This specifies its data type, and reserves memory for it. It

assigns zero to primitive types and null to objects.

dataType variableName;

The second way to declare a variable: This specifies its data type, reserves memory for it, and

puts an initial value into that memory. The initial

value must be of the correct data type.

dataType variableName = initialValue;

The first way to declare two variables: all of the same data type, reserves memory for each.

dataType variableNameOne, variableNameTwo;

The second way to declare two variables: both of the same data type, reserves

Literals:

Literals in Java are a sequence of characters (digits, letters and other characters) that characterize

constant values to be stored in variables. Java language specifies five major types of literals are

as follows:

1. Integer literals

2. Floating point literals

3. Character literals

4. String literals

5. Boolean literals

3.2.2 Identifiers:

Identifiers are programmer-created tokens. They are used for naming classes, methods, variables,

objects, labels, packages and interfaces in a program. Java identifiers follow the following rules:

1. They can have alphabets, digits, and the underscore and dollar sign characters.

2. They must not start with a digit.

3. Uppercase and lowercase letters are individual.

4. They can be of any length.

Identifier must be meaningful, easily understandable and descriptive.

For example:

Private and local variables like ―length‖.

Name of public methods and instance variables begin with lowercase letter like ―addition‖

3.2.3 Keywords:

Keywords are important part of Java. Java language has reserved 50 words as keywords.

Keywords have specific meaning in Java. We cannot use them as variable, classes and method. Operator:

Java carries a broad range of operators. An operator is symbols that specify operation to be

performed may be certain mathematical and logical operation. Operators are used in programs to

operate data and variables. They frequently form a part of mathematical or logical expressions.

Categories of operators are as follows:

1. Arithmetic operators

2. Logical operators

3. Relational operators

4. Assignment operators

5. Conditional operators

6. Increment and decrement operators

Selection Statement:

Selection statement is also called as Decision making statements because it provides the decision

making capabilities to the statements.

In selection statement, there are two types:

if statement

switch statement

These two statements are allows you to control the flow of a program with their conditions.

4.2.1.1 if Statement:

The “if statement” is also called as conditional branch statement. It is used to program execution

through two paths. The syntax of “if statement” is as follows:

Syntax:

if (condition)

{

Statement 1;

Statement 2;

...

}

else

{

Statement 3;

Statement 4;

...

}

The “if statement” is a commanding decision making statement and is used to manage the flow of

execution of statements. The “if statement” is the simplest one in decision statements. Above syntax is

The if…else statement:

Syntax:

If (condition)

{

True - Statement block;

}

else

{

False - Statement block;

}

switch statement:

In Java, switch statement check the value of given variable or statement against a list of case values and

when the match is found a statement-block of that case is executed. Switch statement is also called as

multiway decision statement.

Syntax:

switch(condition)// condition means case value

{

case value-1:statement block1;break;

case value-2:statement block2;break;

case value-3:statement block3;break;

…

default:statement block-default;break;

}

Iteration Statement:

The process of repeatedly executing a statements and is called as looping. The statements may be

executed multiple times (from zero to infinite number). If a loop executing continuous then it is called as

Infinite loop. Looping is also called as iterations.

In Iteration statement, there are three types of operation:

for loop

while loop

do-while loop

4.2.2.1 for loop:

The for loop is entry controlled loop. It means that it provide a more concious loop control structure.

Syntax:

for(initialization;condition;iteration)//iteration means increment/decrement

while loop:

The while loop is entry controlled loop statement. The condition is evaluated, if the condition is true then the

block of statements or statement block is executed otherwise the block of statement is not executed.

Syntax:

While(condition)

{

Statement block; }

do-while loop:

In do-while loop, first attempt of loop should be execute then it check the condition.

The benefit of do-while loop/statement is that we get entry in loop and then condition will check for very first

time. In while loop, condition will check first and if condition will not satisfied then the loop will not execute.

Syntax:

do

{

Statement block;

}

While(condition);  **Jumps in statement**: Statements or loops perform a set of operartions continually until the control variable will not satisfy the

condition. but if we want to break the loop when condition will satisy then Java give a permission to jump

from one statement to end of loop or beginning of loop as well as jump out of a loop.

“break” keyword use for exiting from loop and “continue” keyword use for continuing the loop.