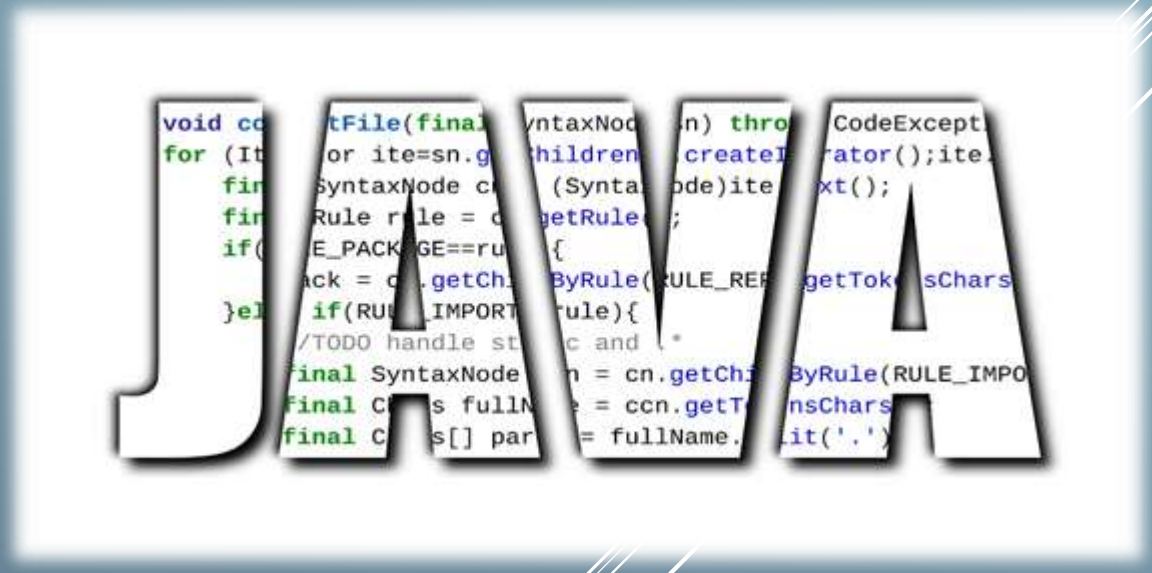


Aazad Waf

What is Java?

Java is a high level programming language originally developed by **Sun Microsystem** but currently owned by **Oracle.**

Short History of Java

James Gosling is the father of Java Language.

Oak changed to **Java** in 1994.

Java programming language public released **1995.**

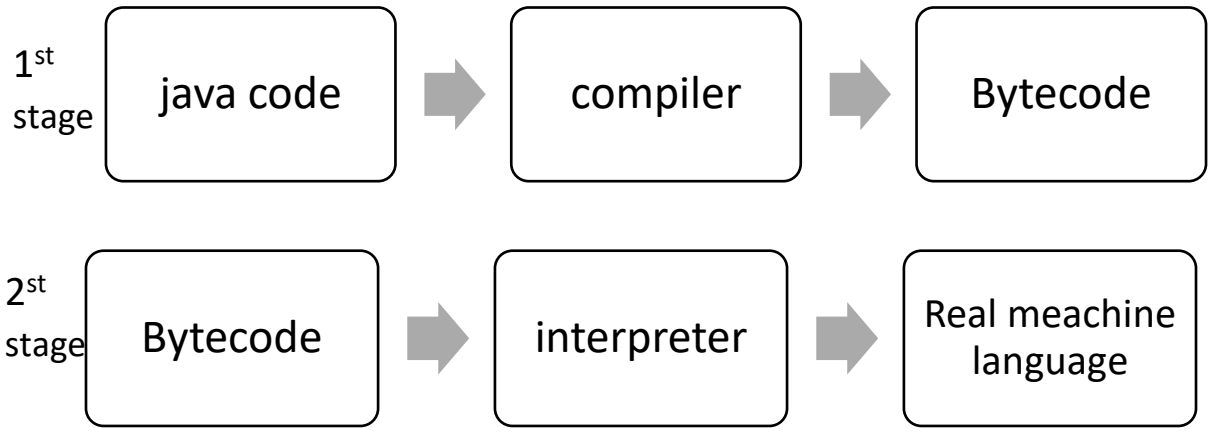
James Gosling

Required Software

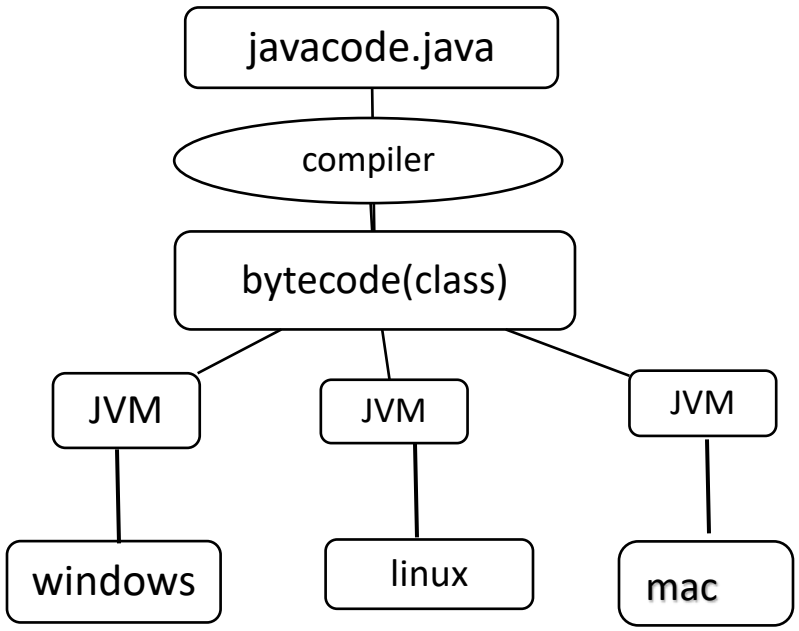
- JDK (Java Development Kit)
- IDE (Integrated Development Environment)
  Netbeans / Eclipse / Jdeveloper

Java Features

- Object oriented
- Compiled and interpreted

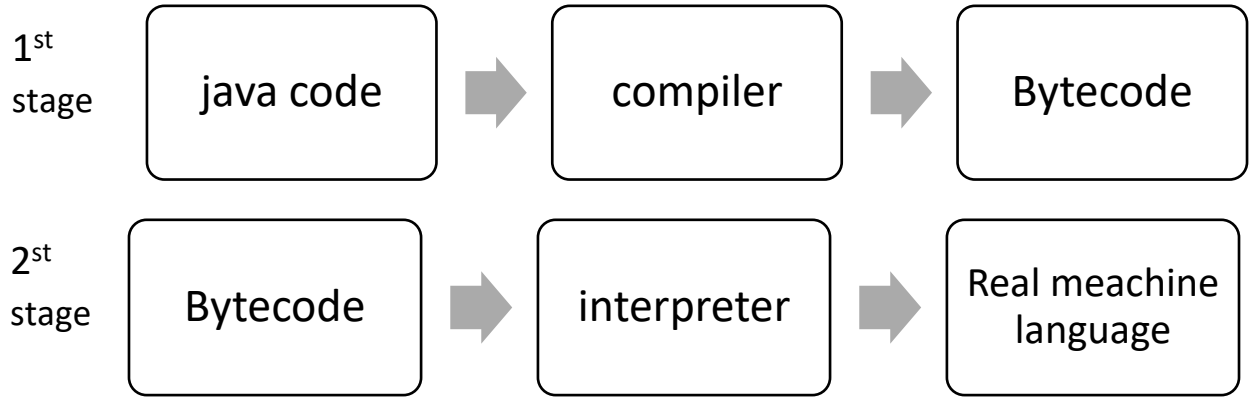


- Platform independent and portable



JVM (Java Virtual Machine)

JVM is a software which work in two stage.



JVM is a kind of software which does **interpreted the Bytecode or class file** and **generate machine code** for real machine language.

Java interpreter is different for each machine that's why different machine generate **machine code** for their each **interpreter.**

- Distributed (support web based application)
- Dynamic
- Robust and Secure
  Best error handling, virus are not run.
- Multi-Threaded (Jump flexibility)

Thread is a small program. Which has the ability to doing many program as a same time.

In a Video player, we see video and audio play as a same time.

Structure Of A Java Program

1. **Documental Section (optional)**

   Comment Section:

   // only one text hide

   /* multiple line text hide */

   /** multiple line hide */

2. **Package Statement (optional)**
3. **Import Statements (optional)**

Import is a keyword

Let, package name test, Class name Student

Than,

```
package test
   Class Student {
import test.Student; }
```

Import test.*;

Means all class file of test package will be imported.

4. **Interface Statement (optional)**

   In a program, **interface statement** only use when doing **multiple inheritance implement.**

5. **Class Definition (optional)**
6. **Main Class Definition (Essential)**

```
{
Main method definition;
}
```

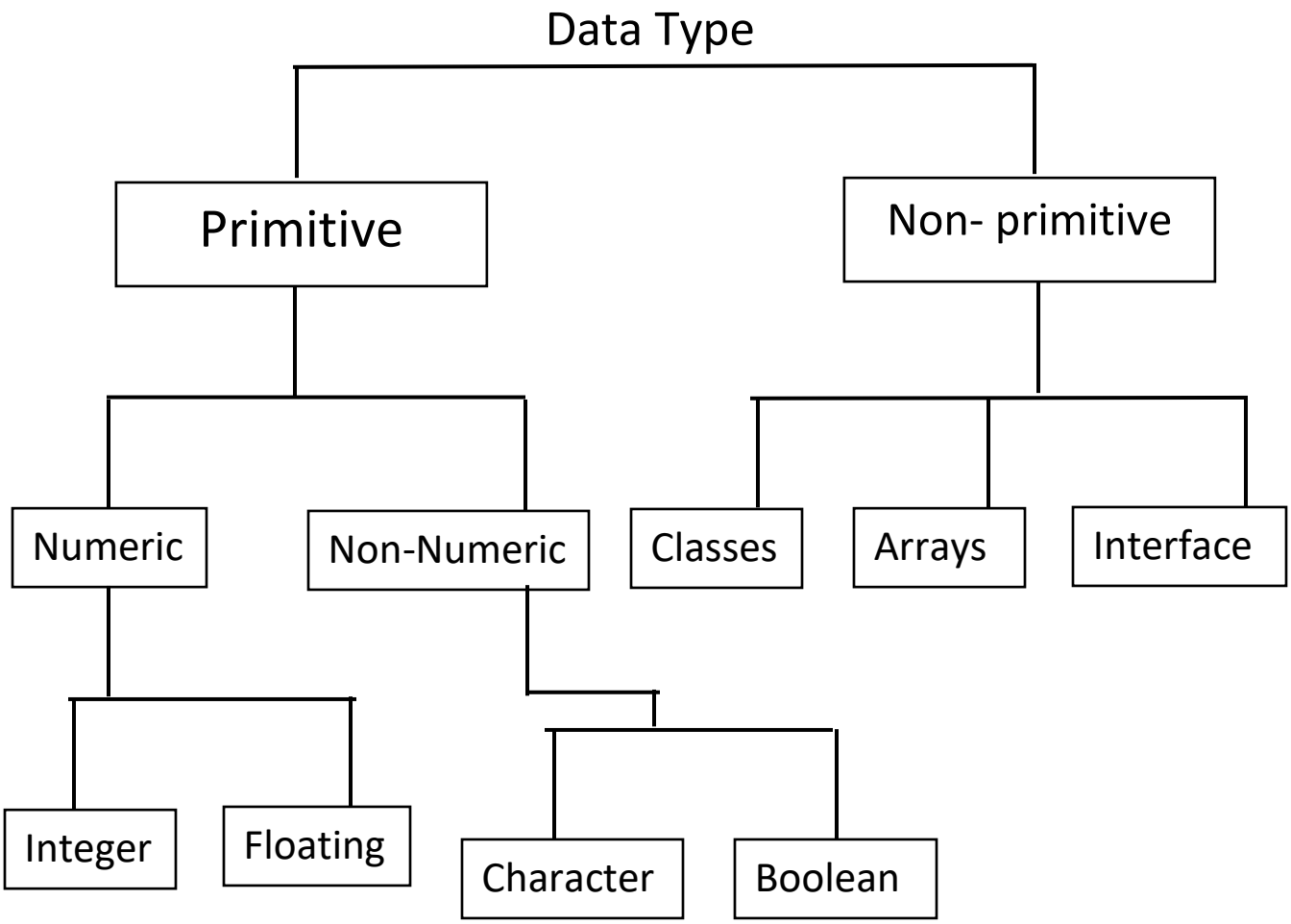
Main method maintained program **entry and exit.**
Program run in a main class and create an **object**.

Data Type

There are 2 kind of data type

i) Primitive data type
ii) Non- primitive data type

Diagram for Data Type



Numeric

Integer (4 types) (memory space in byte)

byte_____ 1 byte
short_____ 2 byte
int_____ 4 byte
long_____ 8 byte

Floating Point (2 types) (memory space)

Float	**4 byte**
Double	**8 byte**

Non-numeric

Char	**2 byte**
Boolean	**1 bit**

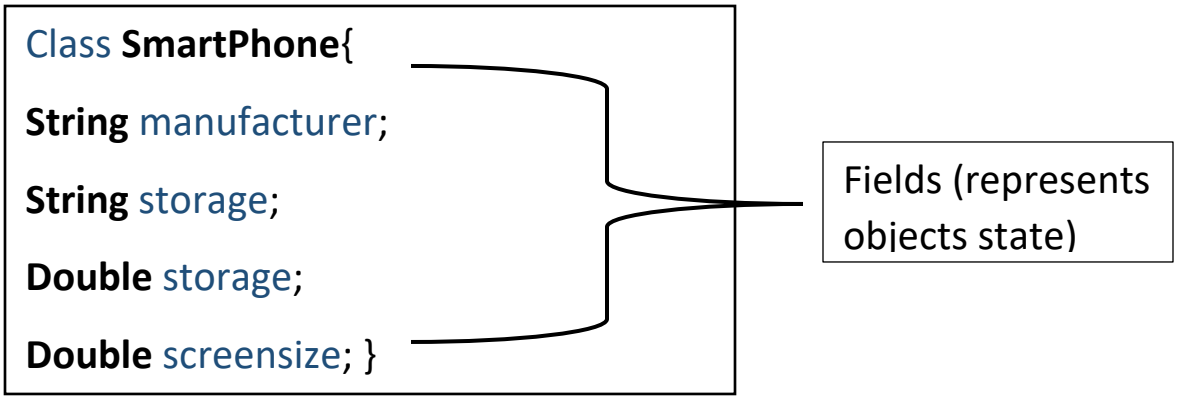
Note: Boolean is not a variable. It use to check true and false value.

Escape Sequence

It is a special character followed by backslash.

Escape sequence		meaning
\b	→	backspace
\t	→	tab
\n	→	new line
\r	→	carriage return
\\ “	→	double quote
\ ‘	→	single quote

Class and Object

- Class is a blueprint from which individual objects are created.
- Classes are logical framework and object physical reality.
- Objects are created using the new operator
- Classes have data and methods, data represents state and methods represents behavior.
- 

- A class is a template (design) for an object and an object is an instance (exhibition) of a class.
- Class define behavior, methods, state of an object.

Define a Class

Optional keyword

[Access modifier] class classname

{

Fields declaration;

Method declaration;

}

Members of class

Note:

In java **Fields are variable** and

Methods are group of statement.

Field Declaration

[Access Modifier] **type** variable name = initial value;

Optional Int, float, etc. Optional

In short: **type** variable name;

Int, float, etc. Any name

Example: float number;

```
Class Test {
Int roll;
Float marks;
Student name;
}
```

Method Declaration

[Access modifier] **return type** **identifier** (parameter list)

Optional — Float, int, etc. — Method name

{

Method body;

}

Example:

```
Class Test {

Null/default   Float   add   (float x, float y)
               Return type  Identifier  Parameter
  {

   Body; }

   }
```

Note:

If method name in 2 words than 1st word 1st letter small and 2nd word 1st letter capital.

Ex. studentName(); getData();

rootMeanSquare();

Program Start

Public static void main (String[] args){

Code statement;}

Public Is a keyword called access modifier. It also be **Private, protected, default.**

Static **Is a** keyword called **static**. If method has static Than in that method, **class** no need to create an object.

Void Is a keyword called **void**. It means no return value Provide.

System.out.println("Text");

System _____ pre-define class

Which has define in java.lang pakage.

Out_____ object

Which is the object of PrintSream class.

println()_____ method

Which is also define in PrintSream class.

A Simple Program

A simple java program given below:

```
package excercise;
public class Simple {
 public static void main(String ar[])
 {
    int a=10, b=20;
    System.out.println("Hello Java");
    System.out.println("Value of A is "+a);
    System.out.println("Value of B is "+b);
 }
}
```

Package name

Class declaration

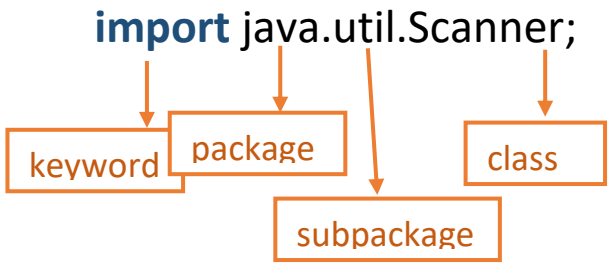
method declaration

Method body;

Scanner

Scanner is a **class**. In java program to take an **input** from user we use **Scanner class**. To take an input first we have to declare **import statement** for **Scanner class**.

Import statement syntax:



Than in a **class** we have to create **object** for **Scanner class.**

Create **Object** for **Scanner:**

Scanner obj = **new Scanner**(**System**.**in**);

For **Integer input** object method is **: nextInt();**

For **Float input** object method is : **nextFloat();**

For **string or line** input object method : **nextLine();**

Example program:

Take an integer input from user and print square of the integer value.

```
package example;
import java.util.Scanner;
public class Square {
   public static void main(String[] args) {
      int a;
      Scanner obj = new Scanner(System.in);
      System.out.println("Enter a number ");
      a = obj.nextInt();
      System.out.println("Square is " + (a * a));
   }
}
```

Decision Making and Branching

Sometimes program has **condition**. If program condition is **true** then execute a **branch or block**. If program condition is **false** then execute **a branch or block**. That type of program will be called **Branching program**.

There are ***three statement*** to branching a program.

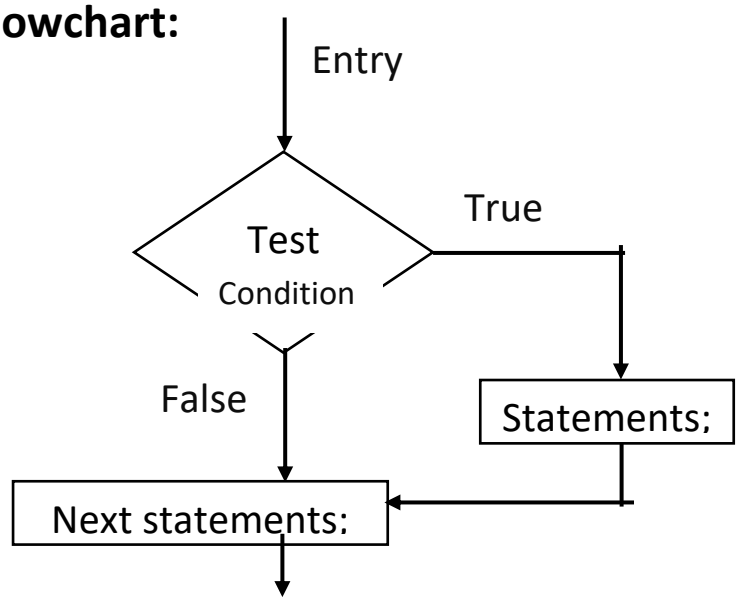
- If statement
- Switch statement
- Condition operator statement

If Statement

If statement **used in 4 way** to branching a program

i) Simple if statement

Simple If statement **flowchart:**



Simple if statement has no **else part**. So if conditions **true** then it follow **statements;** block then **next statements;** and last the program execute. If program condition is **false** then program directly execute the **next statements;** block.

Simple if statement syntax:

```
If(condition){
 statements;
----------; }
Next statements;
```

Example Program:

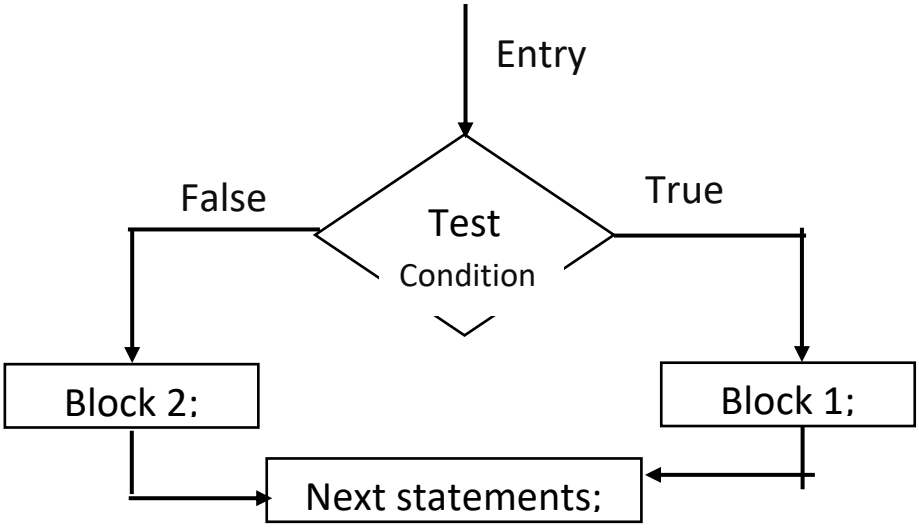
Take an input from user as a salary if

salary>= 10000 taka get 10% bonus. Otherwise no bonus. Print total salary.

```
package example;
import java.util.Scanner;
public class Salary {
   public static void main(String[] args) {
      int sal,b;
      Scanner obj = new Scanner(System.in);
      System.out.println("Enter your salary ");
      sal = obj.nextInt();
      if(sal>=10000){
     b= (sal*10)/100;
     sal= sal+b; }
  System.out.println(“Total salary is” +sal);
    }
}
```

ii) If else statement

If else statement Flowchart:



If else statement has **else part**. It works for both **true** and **false**.

Syntax:

```
If(condition){
 Block 1;
----------; }
else{
 Block 2;
----------; }
Next statements;
```

Example program:

Take an input from user as a salary. If salary>=10000 get 10% bonus. Otherwise get 5% bonus add on salary. Print total salary.

```
package example;
import java.util.Scanner;
public class Salary {
   public static void main(String[] args) {
      int sal,b;
      Scanner obj = new Scanner(System.in);
      System.out.println("Enter your salary ");
      sal = obj.nextInt();
      if(sal>=10000){
     b= (sal*10)/100;
     sal= sal+b; }
   else{
     b= (sal*5)/100;
    sal = sal+b; }
  System.out.println("Total salary is" +sal);
    }
}
```

iii) Else… if ladder statement

iv) Nested if-else statement