Java Course By CodeWithHarry

Java is an Object Oriented programming language developed by Sum Microsystems of USA in 1991
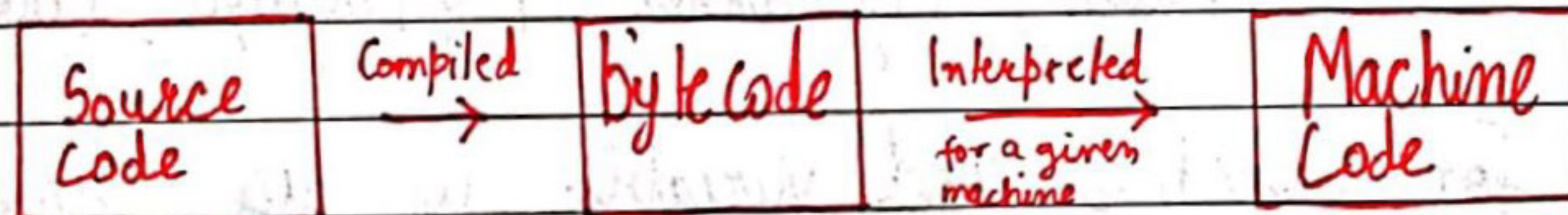
It was originally called Oak by James Goslin
↳ one of the inventors of Java!

JAVA = Purely object Oriented

How JAVA Works?
Java is Compiled into the bytecode and then it is interpreted to machine code

| Source Code | Compiled → | Byte code | Interpreted → for a given machine | Machine Code |

JAVA Installation
Go to Google & type "Install JDK" ⟹ Installs JAVA JDK
Go to Google & type "Install Intellij Idea" ⟹ Installs JAVA IDE

JDK → JAVA Development Kit = Collection of tools used for developing and running Java programs

JRE → JAVA Runtime Environment = Helps in executing programs developed in JAVA

Basic Structure of a Java Program

package com.company; → Groups classes!

Entrypoint into the application

public class Main {
    public static void main (String[] args) {
        System.out.println ("Hello World");
    }
}

※ "Command line argument" baki computer language mai otional hota hai but java mai compulsoru hota hai.

Naming Conventions

→ For classes, we use PascalConvention. First and Subsequent characters from a word are capital letters (uppercase)

Example :
    Main, MyScanner, My Employee, CodeWithHarry

→ For functions and variables, we use CamelCase Convention. Here first character is lowercase and the subsequent characters are uppercase like below:
    main, myScanner, myMarks, CodeWithHarry

Skeleton of JAVA Program

First.java

```
import   java.lang.*;

class   First
{

    public static void  main (String args[])
    {

        System.out.println("Hello World");

    }

}
```

isko import karna optional hota hai, but agar ham nahi bhi likha to automatically import ho jata hai

C:\>javac First.java

First.class

C:\> java First

public static void main(String[] args)

{ ... }

↓ return type

↓ function name

- **public** → access modifier
- **static** → Allows to call this method without creating an object of class
- **void** → return type
- **main** → function name.
- **String[] args** → command line argument

String[] args ↓

[ ] → list

Reading from keyboard

```
import java.lang.*;
import java.util.*;
```

```
class keybRead
{
```

Ya statement aisa hi as it as use karta ha ... input ka liya. Bas "s" ka place par kuch bhi declare kar sakta hai as per rule.

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

    Scanner s=new Scanner(System.in);

    int a,b,c;
    System.out.println("Enter 2 noi");
    a = s.nextInt();
    b = s.nextInt();
    c = a+b;

    S.o.p("sum is" + c);
  }
}
```

Sum is 25

class Scanner

nextInt()

nextFloat()

nextDouble()

next()

nextLine()

nextByte()

nextShort()

nextLong()

nextBoolean()

hasNextInt()
hasNextFloat()

Method	`next()`	`nextLine()`
Description	Reads the next token (word) from the input	Reads the entire line (until newline) from the input
Input Format	Parses the input until the first delimiter (by default, whitespace)	Parses the input until the end of the line (including any whitespaces)
Usage	Useful for parsing individual words or tokens	Useful for reading entire lines of text
Behavior on Spaces	Stops reading at the first whitespace encountered	Reads all characters until the newline, including spaces
Example	If the input is "Hello World", `next()` will return "Hello"	If the input is "Hello World", `nextLine()` will return "Hello World"

```java
import java.util.Scanner;
public class wlc
{
    Run | Debug
    public static void main (String[] arg )
    {
        Scanner sc=new Scanner(System.in);
        System.out.print(s:"Enter your name = ");
        String name;
        name = sc.nextLine();
        /*ya par nextLine ka use iseliya kiya hai taki ham
          ek sa jayda word ka print kar saka */
        System.out.println("Welcome "+name);
    }

}
```