



Internet Programming

BCA503-Introduction to Internet Programming

UNIT-1

(Lotus Institute)

By

Shadab Ali

Research Scholar (CSIT)

MJP Rohilkhand University



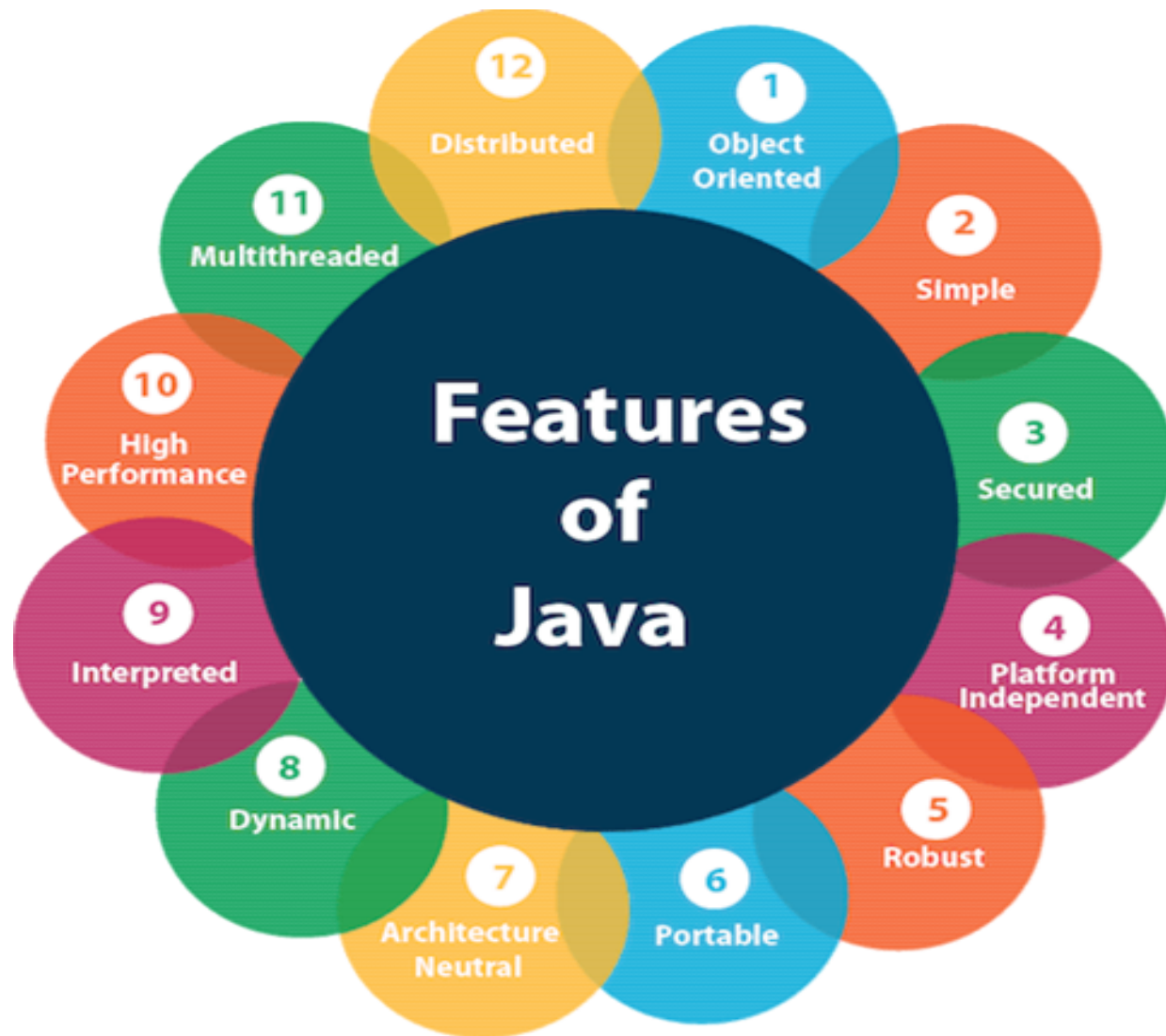
What is an Internet Programming?

Java Programming Language Overview

Java is a widely-used programming language known for its versatility and power. Created by **Sun Microsystems** in **1995** (now *owned by Oracle*), its main feature is **platform independence**. This means Java programs can run on any device with a **Java Virtual Machine (JVM)**, no matter the operating system (Windows, Linux, Mac).

Features of Java:

- ✓ **Simple and Easy to Learn:** *removes complex features like pointers and manual memory management*
- ✓ **Object-Oriented:** *Java is built around the concept of **Objects** and **Classes***
- ✓ **Platform-Independent:** *Write Once, Run Anywhere*
- ✓ **Secure:** *has built-in security features that protect applications from viruses*
- ✓ **Multithreading:** *allowing multiple tasks (threads) to run simultaneously*
- ✓ **Robust and Reliable:** *Garbage Collection & extensive error-handling mechanisms*



Java Program Structure

A basic Java program has the following structure:

java

Copy code

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

- **Class:** Every Java program is defined inside a class (here, `HelloWorld`).
- **Main Method:** This is the starting point of every Java application. The code inside `main()` is executed when the program runs.
- **System.out.println():** This command prints text (here, "Hello, World!") to the console.

- **JDK (Java Development Kit):** Contains tools needed for developing Java applications, including the **compiler** (to compile code) and the **JVM** (to run it).
- **JRE (Java Runtime Environment):** A subset of the JDK that includes the JVM and libraries needed to run Java applications.

Applications of Java:

Desktop Applications: *Java can be used to create standalone applications with graphical interfaces*

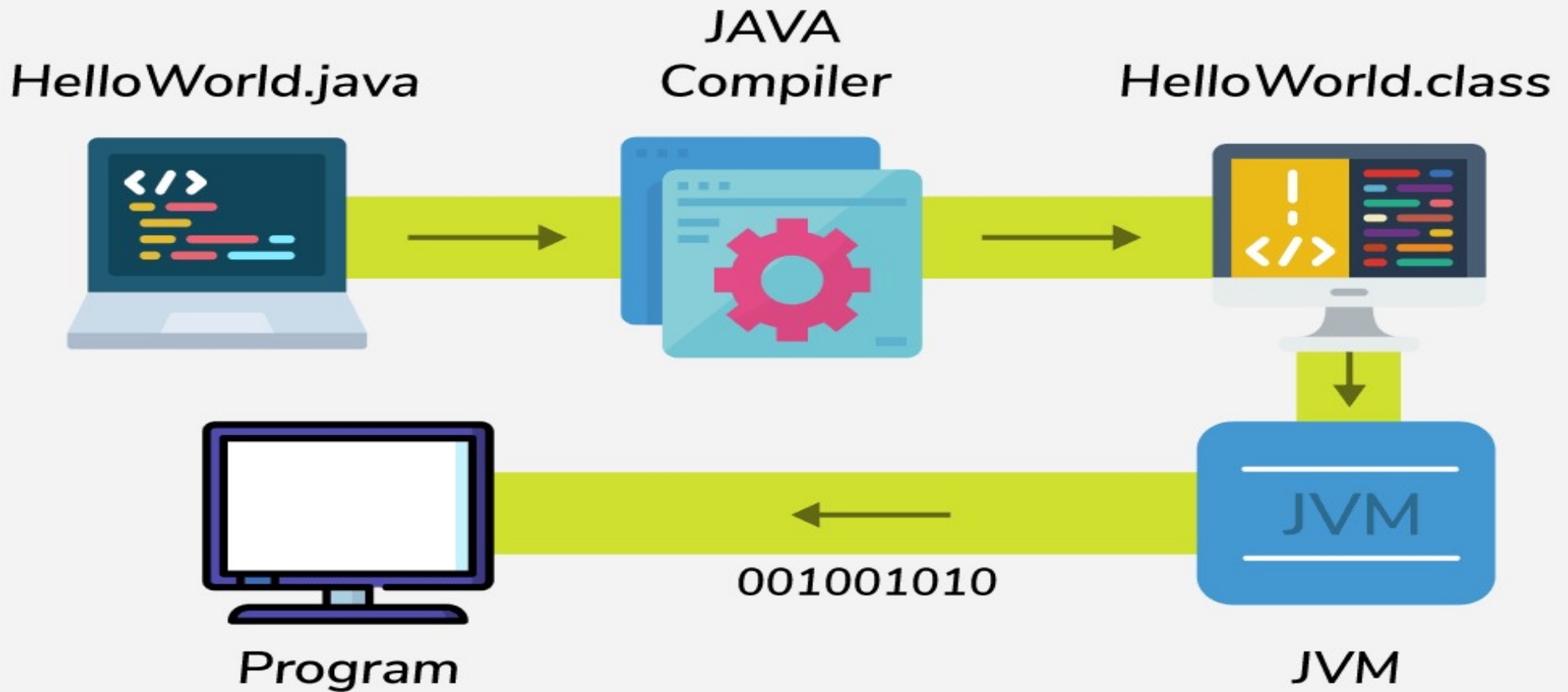
Web Development: *Java is used for building dynamic web applications with frameworks like **Spring & JSP***

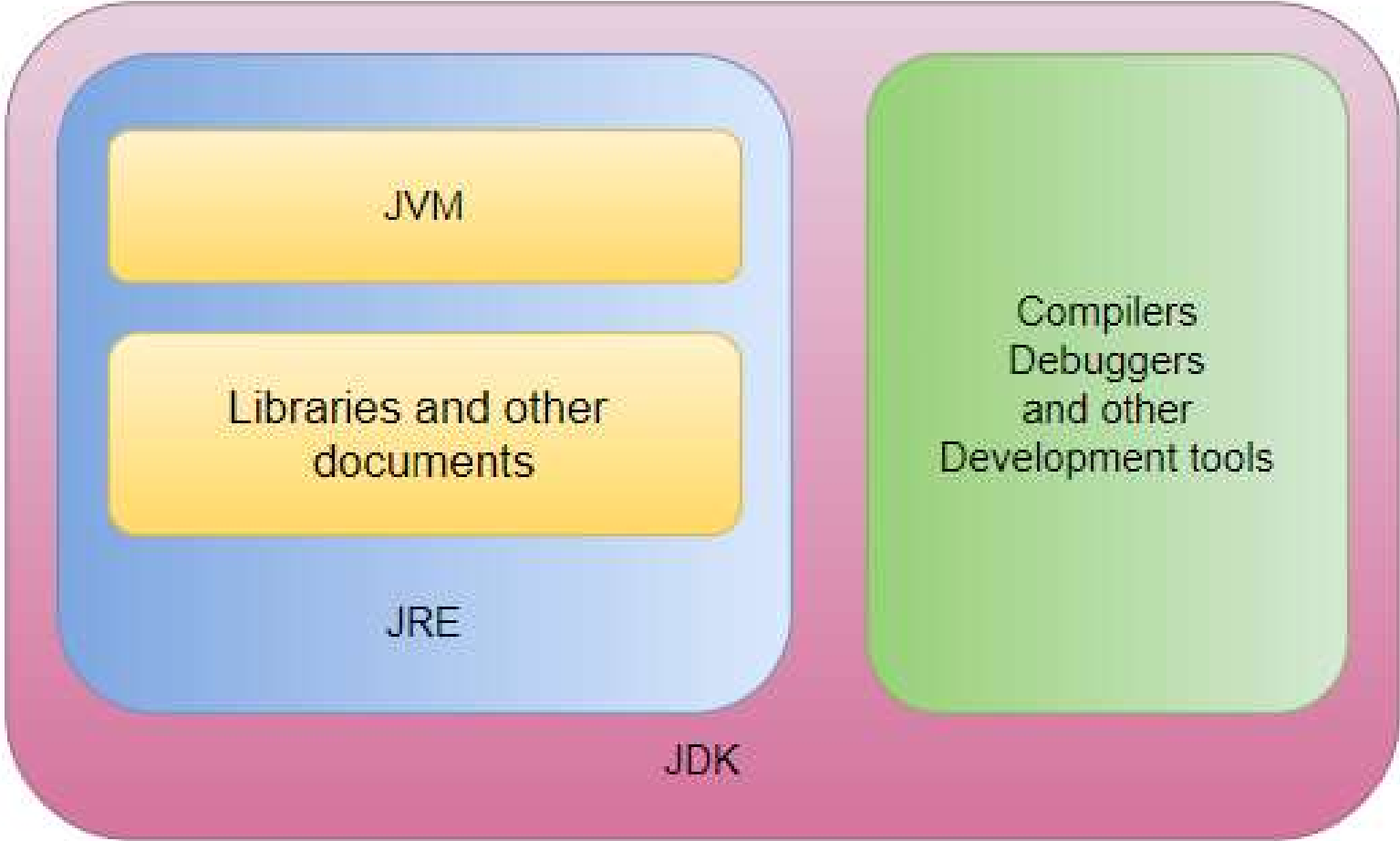
Mobile Applications: *Java is the primary language for Android app development*

Enterprise Applications: *Java is widely used in large-scale systems, like banking and e-commerce platforms*

Games and Cloud Computing: *Java can be used to develop video games and cloud-based applications*

How Java Code Executes





Referring to applets and Applications

Key Differences Between Applets and Applications:

Aspect	Java Applications	Java Applets
Execution	Runs independently on any OS with JVM	Runs inside a web browser (or AppletViewer)
Main Method	Has a <code>main()</code> method for execution	No <code>main()</code> , uses lifecycle methods like <code>init()</code>
Security	Can access system resources like files	Restricted by the browser's security sandbox
Deployment	Distributed as JAR files or standalone executables	Embedded in web pages
Status	Actively used in Java development	Deprecated and not supported in modern browsers

First Step in Writing Java Application

1. Install the JDK
2. Choose a text editor or IDE
3. Write a simple Java program (HelloWorld.java)
4. Compile the program using javac
5. Run the program using java

Primary Application Components

1- Class:

A class is the blueprint or template that defines the fields and methods of objects.

```
public class Car {  
    // Fields  
    String model;  
    int year;  
  
    // Methods  
    public void start() {  
        System.out.println("The car has started.");  
    }  
}
```

Primary Application Components

2- Object:

An object is an instance of a class. It is created using the *new* keyword and represents a specific instance with real data.

```
Car myCar = new Car(); // Create an object of the Car class  
myCar.start();      // Call the start() method
```

3- Method:

A method is a block of code that performs a specific task. Methods are used to define the behavior of objects.

```
public void drive() {  
    System.out.println("The car is driving.");  
}
```

Primary Application Components

4- Constructor:

A constructor is a special type of method used to initialize objects when they are created has the same name as the class and does not have a return type.

5- Main Method:

The main() method is the entry point of any Java application.

When a Java program is run, the JVM looks for the main() method to start the execution.

6- Variables:

Variables are used to store data within a program.

Class Code Block

A class code block is the area inside a class where you define the fields (variables), methods, constructors, and other members of the class. It forms the structure of the class and dictates how it behaves when objects are created from it.

Method Code Block

A block of code that defines a specific task or behavior that can be performed by objects of a class or the class itself. A method usually consists of the method's name, parameters (if any), and the logic or task that it performs.

```
public int addNumbers(int a, int b) {  
    // Body of the method where the logic is written  
    int sum = a + b; // Calculate the sum of two numbers  
    return sum;     // Return the result  
}
```

Java Comments

Comments in any programming language are ignored by the compiler or the interpreter. A comment is a part of the coding file that the programmer does not want to execute, rather the programmer uses it to either explain a block of code or to avoid the execution of a specific part of code while testing.

There are two types of comments:

- **Single-line comment**

To write a single-line comment just add a `'//'` at the start of the line.

- **Multi-line comment**

To write a multi-line comment just add a `'/*.....*/'` at the start of the line.

Using Semicolons and braces

1. Semicolons (;):

- ✓ A semicolon marks the **end of a statement**.
- ✓ It's used to separate instructions so that the Java compiler knows where one instruction ends and the next begins.

int a = 5; // Declaration of a variable

System.out.println(a); // Print statement

- Without a semicolon, Java will not understand where the statement ends, causing a syntax error

2. Braces ({ }):

- ✓ Braces define a block of code. They are used to group multiple statements.

We use Braces in following:

Classes: To enclose the code of a class.

Methods: To define the body of a method.

Loops and Conditionals: To group the statements that should execute in response to conditions or loops.

Compiling and running a Java Program

1. Write the Java Program:

- ✓ Save the file with a .java extension (ex. HelloWorld.java)

2. Open the Terminal/Command Prompt:

- ✓ Navigate to the folder where your .java file is saved

3. Compile the Program:

By running the command as: `javac HelloWorld.java`

- ✓ This generates a bytecode file called HelloWorld.class

4. Run the Program:

- ✓ Execute the program using: `java HelloWorld`

- Don't include .class in the command

Requirements for Your Source File

1. File Name and Class Name:

- ✓ If you have a public class, the file name must match the class name exactly
- ✓ For example: *If the class is HelloWorld, the file name must be HelloWorld.java*

2. File Extension:

- ✓ The source file must have a .java extension , *for example: HelloWorld.java*

3. One Public Class per File:

- ✓ A source file can only have one public class
- ✓ The public class name must match the file name

4. Proper Structure:

- ✓ Optional: Package declaration at the top
- ✓ Optional: Import statements for libraries
- ✓ Required: Class definitions with the main method if it's the main program

Compiling & Running

Compiling:

The process of compiling a Java program involves converting human-readable code (source code) into bytecode that can be understood by the Java Virtual Machine (JVM). This is done using the Java compiler.

1. Writing the Source Code
2. Using the javac Command
3. Generating Bytecode
4. Handling Compilation Errors

Running:

Once your Java program has been compiled into **bytecode**, you can execute it using the **Java Virtual Machine (JVM)**. The JVM interprets the bytecode and runs your program.

1. Ensure the Program is Compiled
2. Use the java Command
3. Program Output
4. Error Handling