Unit-1
Introduction To Java

Ms. Nehal Adhvaryu

Java:

Java is a general-purpose, object-oriented programming language developed by Sun Microsystems of USA in 1991.

Features Of Java:

1. Simple:

Java was designed to be easy for the professional programmer to learn and use effectively.

If you already understand the concept of object-oriented language, it is easier to learn java.

2. Object-Oriented:

Java is a pure object-oriented language. Almost everything in java is an object.

OOPs is an approach that provides a way of modularizing programs by creating partitioned memory area for both data and function that can be used as templates for creating copies of such module on demand.ss

3. Robust:

It provides many safeguards to ensure reliable code. It has strict compile time and runtime checking for data types.

It also incorporate the concept of exception handling which captures series errors and eliminate ant risk of crashing the system.

Security becomes an important issue for a language that is used for programming on internet. Threat of viruses and abuse of resources are everywhere, java system not only verify all memory access but also ensure that no viruses are communicated with an applet. The absence of pointers in java ensures that programs cannot gain access to memory locations without proper authorization.

4. Multi Threaded:

Java was designed to meet the real world requirement of creating interactive networked programs.

To accomplish this, java supports multithreaded programming, which allow you to write program that do many things simultaneously.

5. Architectural Neutral:

Java programs easily moved from one computer system to another, anywhere any time, changes and upgrades in o.s processors and system resources will not force any changes in java programs.

6. Distributed:

Java is designed for the distributed environment of the internet because it handled TCP/IP protocols.

It also support Remote Method Invocation (RMI). This features enables a program to invoke method across a network.

7. Dynamic:

Java is a dynamic language designed to adapt to an envolving environment.

java program carry a lot of runtime information to validate and access objects at runtime. This makes it possible to safely link code dynamically.

Difference Between Java And C

Java	C
1. Java Is an object Oriented Language.	1. C is a structured Language.
2. Java does not support explicit pointer type.	2. C support.
3.Java does not support.	3. C support preprocessor.
4. Java does not contain the data type structure and union.	4.C contain the data type structure, union, and enumeration.
5. Java does not define the type modifiers keywords auto, extern, register, signed and unsigned.	5. C define the type modifiers keywords auto, extern, register, signed and unsigned.
6. Java does not include keywords like sizeof and typedef.	6. C include.
6.Java add new operator such as instanceof and >>>.	6. C does not have this type of operator.

Difference Between Java And C++.

Java	C++
1. Java is a truly object-oriented language.	1. C++ is basically c with object-oriented extension.
2. Java does not support operator overloading.	2. C++ support.
3. Java replace the destructor function with finalize() function.	3. C++ has destructor.
4.There are no header file in java.	4. C++ have header file.
5. Java does not support multiple inheritance.	5. C++ support multiple inheritance.
6.Java does not support global variable every variable and method is declared within a class.	6. C++ support global variable.
7. Java does not use pointer.	7. C++ use pointer.

Java And Internet

Java is strongly associated with the internet because of it's three features.

1. Java Applets

An applet is a special kind of java program that is designed to be transmitted over the internet and automatically executed by java-compatible browser.

An applet is downloaded on demand, just like an image, sounding file or video clip. The difference is that an applet is an intelligent program not just an animation or media file.

2. Security

When you use a java compatible web browser, you can safely download java applets without fear of viral infection or malicious intent.

Java achieves this protection by confining a java program to the java execution environment and not allowing it access to other parts of the computer.

3. Portability

Java Program have ability to run on different platform without modifications.

Bytecode

The key that allow java to solve both the security and the portability problems.

Bytecode is a highly optimized set of instructions designed to be executed by the java runtime system called as java virtual machine.

The output of a java compiler is not executable code is called bytecode.

Process Of Compilation

Java Virtual Machine (JVM)

JVM is a java runtime system, which is an interpreter for bytecode.

Translating a java program into bytecode makes it much easier to run a program into wide variety of environment because only the JVM needs to be implemented for each platform.

Processing Of Converting Bytecode Into Machine Code

Just-In Time (JIT)

- JIT is a technique for improving the runtime performance of a computer program.

-Traditionally, computer programs had two modes of runtime operation, either interpreted or static compilation.

Interpreted code was translated from a high-level language to a machine code continuously during every execution, whereas static compilation translates code into machine code before execution, and only required this translation once.

-JIT compilers represented a hybrid approach, and used for bytecode.

-when JIT is a part of a JVM, selected portion of bytecode are compiled into executable code, it is not possible to compile the whole program into executable code all at once, because java perform various runtime check that can be done only one at run time, during execution, JIT compiler compile the code as it needed.

Java Environment:

Java environment includes a large number of development tools and hundreds of classes and methods.

The development tools are part of the system know as Java Development Kit (JDK) and the classes and methods are part of the Java Standard Library (JSL), also know as the Application Program Interface (API).

Java Development Kit (JDK)

The Java Development Kit comes with a collection of tools that are used for developing and running java program.

They include:

- appletviewer for viewing applet
- javac (Java compiler)
- java (Java Interpreter)
-javap (Java disassembler)
-javah (for c header file)
-javadoc (for creating HTML document)
-jdb (Java debugger)

Application Programming Interface (API)

The java Standard Library or API includes hundreds of classes and methods grouped into several functional packages. Most commonly used packages are:

Language Support Package: for implementing basic features of java.

Utilities Package: provide utility function such as date & time function.

I/O Package: for i/o manipulation.

N/W Package: for communicating with other computers via Internet

AWT Package: abstract window tool kit, that implement platform-independent graphical user interface

Applet Package: allows us to create java applets.

Overview Of Java Language:

Java is a general purpose , object-oriented language. We can develop two types of java programs

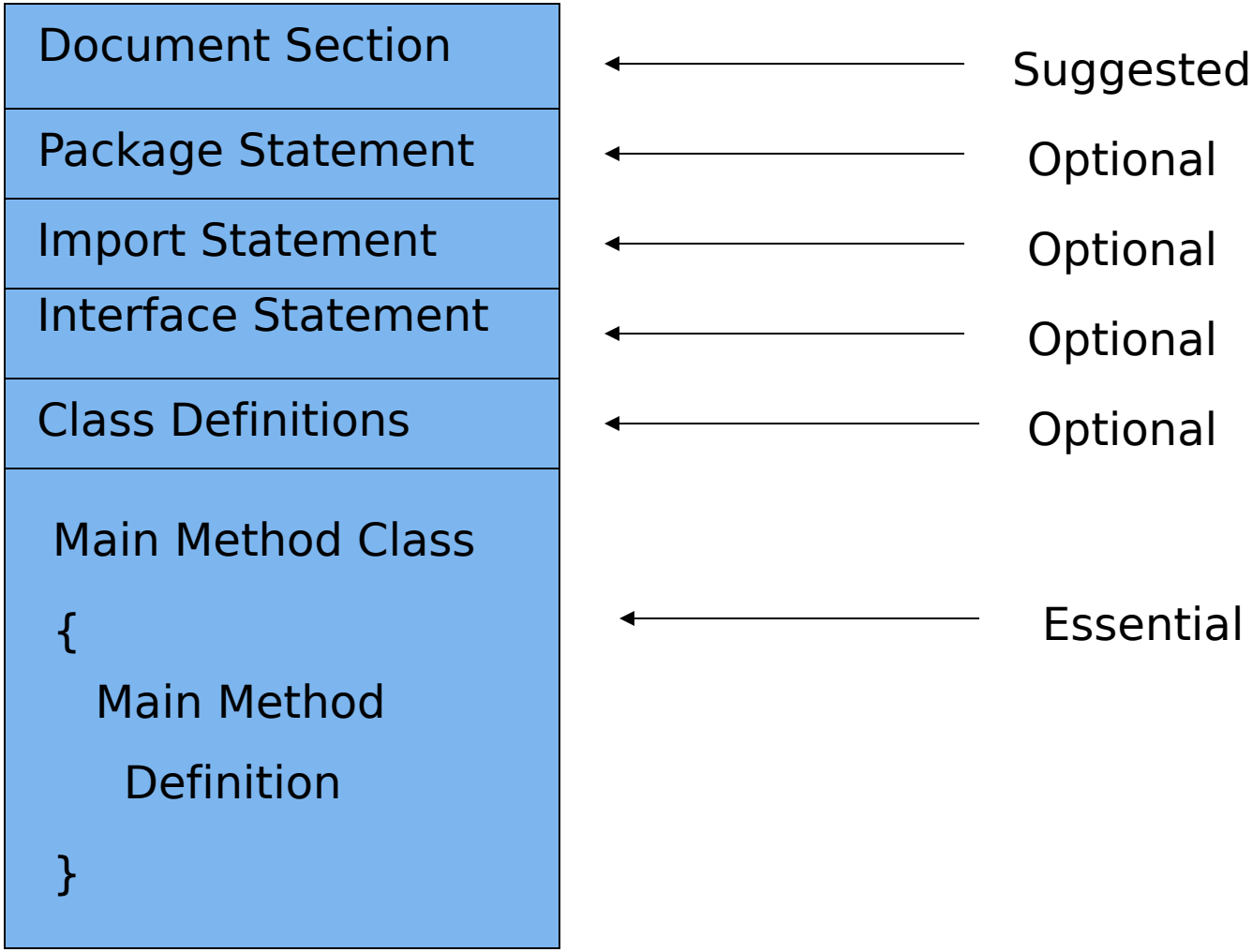
- Stand-alone application:
  - Stand-alone application are programs written in java to carry out certain tasks on a stand-alone computer.
  - Executing a stand-alone alone java program involve two steps
    1. Compiling source code into byte code using javac compiler.
    2. Executing the bytecode program using java interpreter.

:

- Web applets:
  - Applets are small java program develop for internet application.
  - An applet located on a distance computer server can be downloaded via Internet and executed on a local computer using java capable browser.

Creating and running applets are more complex than creating an application.

Java Program Structure:



Implementing Java program:

Implementation of a java application program involves a series of steps. They include:

1. Creating the program
2. Compiling the program
3. Running the program

1. Creating the program:

We can create a program in any text editor. The file name must be the class name of the class containing the main method and extension is .java. This file is called as the source file.

2. Compiling the program:

To compile the program we must run the java compiler javac, with the name of the source file on the command line. If every thing is ok the javac compiler creates a file called sourcefilename.class containing the bytecode of the program.

3. Running the program:

We need to use java interpreter java, to run a stand-alone program.

Example Of Java Program:

```
/*
This is a simple Java program.
Call this file "Example.java".
*/
class Example {
        // Your program begins with a call to main().
        public static void main(String args[]) {
                System.out.println("This is a simple Java program.");
                }
        }
```

Main method:

public static void main(String args[])

This line define a method named main. This is the starting point for the interpreter to begin the execution of the program.

A java program can have any number of classes but only one of them must include a main method to initiate the execution.

This line contain a number of keywords, public, static and void.

public:

It is an access specifier which make it accessible to all other class.

static:

The main must always be declared as static since the interpreter use this method before any objects are created.

void :

The type modifier states that the main method does not return ant value.

String args[ ]:

It declare parameter named args, which contain an array of objects of the class type String.

It is used for command line argument, that are supplied to the application program at the time of invoking it for execution.

Any argument provided in the command line are passed to the array args as its elements.

The Output line:

System.out.println ("This is a simple Java program.");

Here println method is a member of the out object,which is a static data member of the System class.

The println always appends a newline character to the end of the string.