Java means DURGA SOFT..

CORE JAVA

Material

Java™

India's No.1 Software Training Institute

DURGASOFT

www.durgasoft.com Ph: 9246212143 ,8096969696

1. Introduction

Java was Introduced in 1996 January.But it was started by Sun MicroSystems in 1991. At that time Sun MicroSystems had a requirement to prepare a software electronic consumer devices in 1991.They need a programming language to prepare those devices.30 members team with James Gosling & patrickNaughton starts "Green" project.For this project they need a simple programming language that is simple,tight coded and architectural neutral programming language.

How can we justified that programming language is Simple....?

Ans:The electronic consumer devices with less memory,less power,less execution time the programming language which is used to prepare these devices must be simple programming language.

For Example:I want to prepare stack operation in "C"

Stack Output:

1.push ----> 15 to 20 lines of code we need

2.pop ----> 15 to 20 lines of code we need

3.peep ----> 15 to 20 lines of code we need

Total ----> 45 to 60 lines of code we need

JAVA:

To perform stack operations.

Ex:

```
Stack s=new Stack();

s.push("A");

s.pop();

s.peep();
```

---

4 lines of code we need to write stack operations in Java

To write less no.of instructions in a programming language,then that is called as tight coded programming language.It achieving more complex requirements.

Q)What is Architectural Neutral?

Ans:Number of electronic devices manufactures are available in now a days like Panasonic,LG,Sony,Samsung...etc.They have their own remote controllers.Each remote controller have their own hardware Architecture.These architecture differ from manufacturer to manufacturer.If they want my software,after the demonstration,then I sell my product to Sony Company(If it accept my Quotation).

If my software is compatible to,then there is no possible to run my software on H2 and H3 hardware devices.Then my business is down.If I wanna to go to improve my business,my software product should not compatible to only one hardware i.e my software is hardware independent.It increases my business.It is called "hardware neutral software" in another way.

To prepare this hardware independent software,I used programming language.This programming language should be common to all hardware systems.That programming language is called as independent "Architectural Neutral Programming language".

Operating System independent ===> platform Independent.

In 1991,one programming language is famous at that time i.e PASCAL it is implemented by "Niclous Wirth" (Invented of Pascal).

He gave a software to James Gosling on the base of PASCAL programming language(Procedure Oriented programming language).But James Gosling expected object oriented programming language.So he rejected the solution given by Niclous Wirth.

Because all the team members came from Object Oriented background.Because of this mismatch they rejected the PASCAL Programming Language

solution.So in 1992,December,James Gosling and his team create a new programming language.

James Gosling always likes to see a "Oak Tree" through his cabin window.Immediately he thought to put this "Oak" name for his new programming language.But his team rejected this name.Because this "Oak PL" is already existed.So all the Green Project people think a new name for a new Programming language.

While they are doing Green project,they used to go a cafeteria in Sun MicroSystems.They drank a Coffee daily that is named as "JAVA".So they decided to keep "JAVA" is a name for new programming language.

Java is one of Island in Indonesia.Java is famous for coffee trees.So that symbol for Java T.B is Coffee cup and a saucer with flames.

Dec 1992:

The first product prepared on Java is "*7".It is a remote controller.But unfortunately,no customer had an internet to take this product upto 1994.It is a failure product.

In the middle of 1994,very good boom is available for "www" internet.People are identified a key role for www is "browser".The name of this browser is "Musaik".It is prepared by "Mark Anderson".He is an under graduate student.He went to Netscape communication to prepare "Musaik Browser".He got a stipend 6.25$ per hour at Netscape Communication.

Then Sun MicroSystems decided to prepare its own browser.So patrik Naughton and Janadhan paine in 1995,Dec prepared "Hot Java" browser using Java programming language.

Sun MicroSystems got very good reputation for this "Hot Java Browser". The key concept for this browser is "Applets" so in January 23,1996--->JDK 1.0 version released.

Now Java 8.0 version is introduced in Mar 2014

Java Details:

Home	SUN Mc Systems.[Oracle Corporation]
Author	James Gosling.
Objective	To prepare simple electronic consumer goods.
Project	Green
First Version	JDK1.0 [1996,Jan]
Used version	some org JDK5.0, some other JAVA6
Latest Version	JAVA7,JAVA8
Type of Software	open source software.
Strong Features	Object-oriented, Platform Independent, Robust, Portable, Dynamic, Secure..

Java Evolutions:

Version	Code Name	Enhancements
1.JDK1.0[Jan,23,1996]	OAK	Language Introduction
2.JDK1.1[Feb,19,1997] RMI,JDBC,Reflection API	-----	Java Beans,Inner classes
3.JDK1.2[Dec,8,1998]	Playground	Strictfp,Swing,CORBA, Collection Framework
4.JDK1.3[May,8,2000]	Kestrel	Updations on RMI,JNDI
5.JDK1.4[Feb,6,2002]	Merlin	Regular Expression, NIO, assert keyword,JAXP,...
6.JDK5.0[Sep,30,2004]	Tiger	Autoboxing, var-arg method, static import,Annotations,..
7.JAVA SE6[Dec,11,2006]	Mustang	JDBC4.0,GUI updations, Console class.
8.JAVA SE7[July,28,2011]	Dolphin	Strings in switch '_' symbol in literals try-with-resources
9.JAVA SE8 [Mar,18,2014]	spider	interface improvements Lambda Expression Date-Time API Updations on Collections

Differences between Java and Others[C and C++]

1.C and C++ are Static programming Languages But Java is a dynamic programming language:

If any programming language allows memory allocation at compilation time then that programming language is called as Static programming language.

Ex: C and C++

In C and C++,memory will be allocated for the primitive data Types at compilation time only.

If any programming language allows memory allocation at runtime then that programming language is called as Dynamic Programming language.

Ex: JAVA

In Java applications,memory will be allocated for the primitive dataTypes at runtime only,because,in Java applications objects will be created at runtime.

2.C and C++ are required pre-processor but Java is not required pre-processor:

In case of C and C++,the complete predefined library in the form of header files.

Ex:stdio.h

conio.h

math.h

If we want to include header files in C and C++ applications we have to use "#include<>" statements.

Ex: #include<stdio.h>

#include<conio.h>

#include<math.h>

If we compile C and C++ application,pre-processor will recognize #include<> statements and pre-processor will load the specified header files content to the memory at compilation time, this type of loading is called as "Static loading".

In C and C++ applications,it is mandatory to use header files and #include<> statement so that pre-processor is required in C and C++.

In case of Java,the complete predefined library is provided in the form of "packages".

Ex: java.util

java.io

java.net

In Java applications,if we want to include packages then we have to use "import" statement,not #include<> statement.

Ex: import java.util.*;

import java.io.*;

import java.net.*;

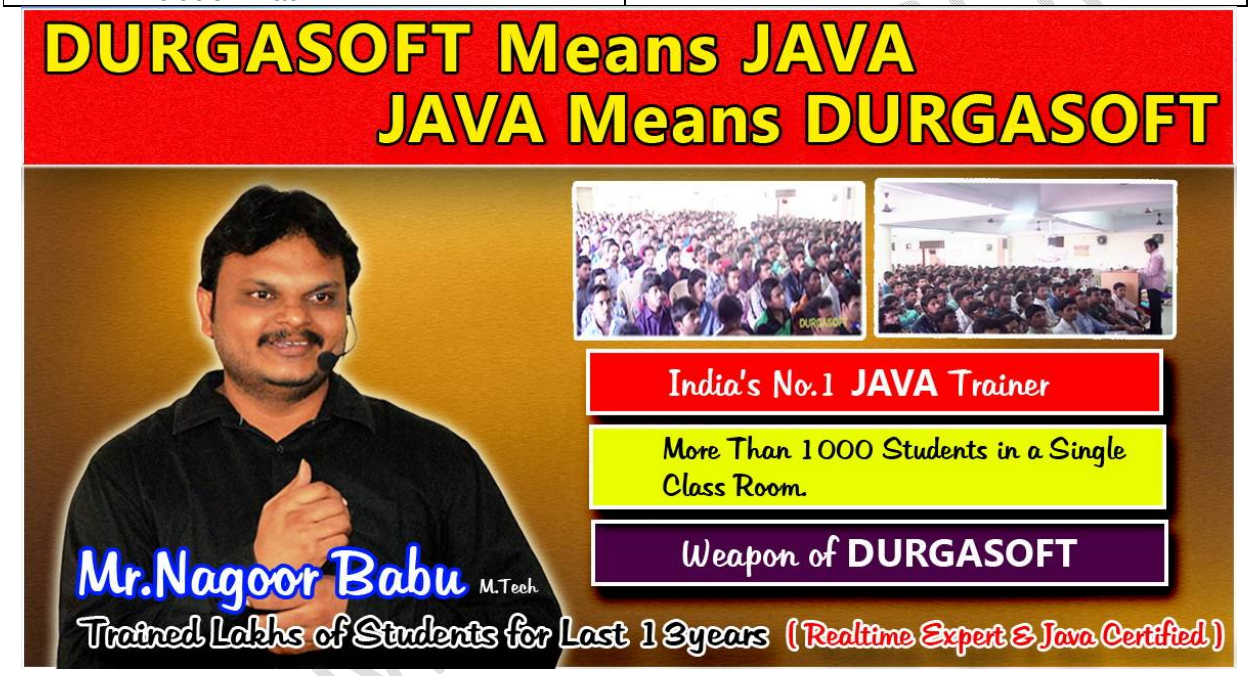
when we compile java files compiler will recognize import statements, where compiler will check whether the specified packages are available or not,if not, compiler will rise an error like "package java.util does not exist".If the specified package is available in java predefined library then compiler will not load package content to the memory, at runtime JVM will load the required classes or interfaces of the specified package, this type of loading is called as "Dynamic Loading".

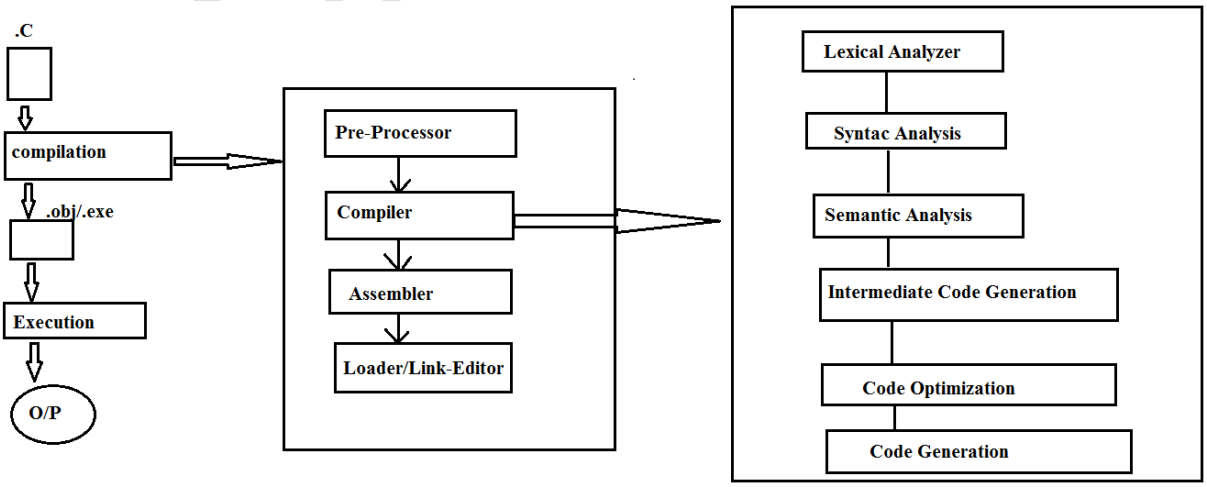
In Java, include<> statements are not available, so that,Pre-Processor is not required in Java.

Q)What are the differences between #include<> statement and "import" statement?

1.#include<> statement is available in C.	import statement is available in Java.

2. #include<> statement can be used to include the predefined library specified with header files.	import statements can be used to include the predefined library specified with packages.
3. #include<> statements will be recognized and executed by the pre-processor.	import statements will be recognized by the compiler.
4. #include<> statement will provide static loading.	import statement will provide dynamic loading.
5. If we want to include more than one header file in C and C++ applications then we have to use more than one #include<> statements. Ex: #include<stdio.h> #include<conio.h> #include<math.h>	If we want to include more that one class or interface of the same package then it is sufficient to use single import statement with '*' notation. Ex:import java.util.*;





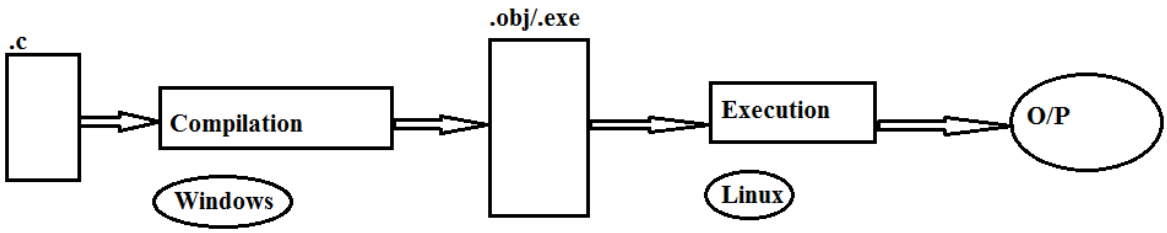
Internal Mechanism of 'C' Compilation

3.C and C++ are platform Dependent but Java is platform independent:

If any programming language allows its applications to compile and to execute on the same operating system then that programming language is called as "Platform Dependent Programming Language".

Ex: C and C++

If we compile C program on windows operating system then compiler will generate .obj or .exe file with the same windows operating system representations.If we want to execute C program then we must require the same windows operating system,this nature of C programming language is "Platform Dependent Programming Language".



Compilation of "C"

If any programming language allows its applications to compile on one operating system and to execute on another operating system then that programming language is called as "Platform InDependent Programming Language".

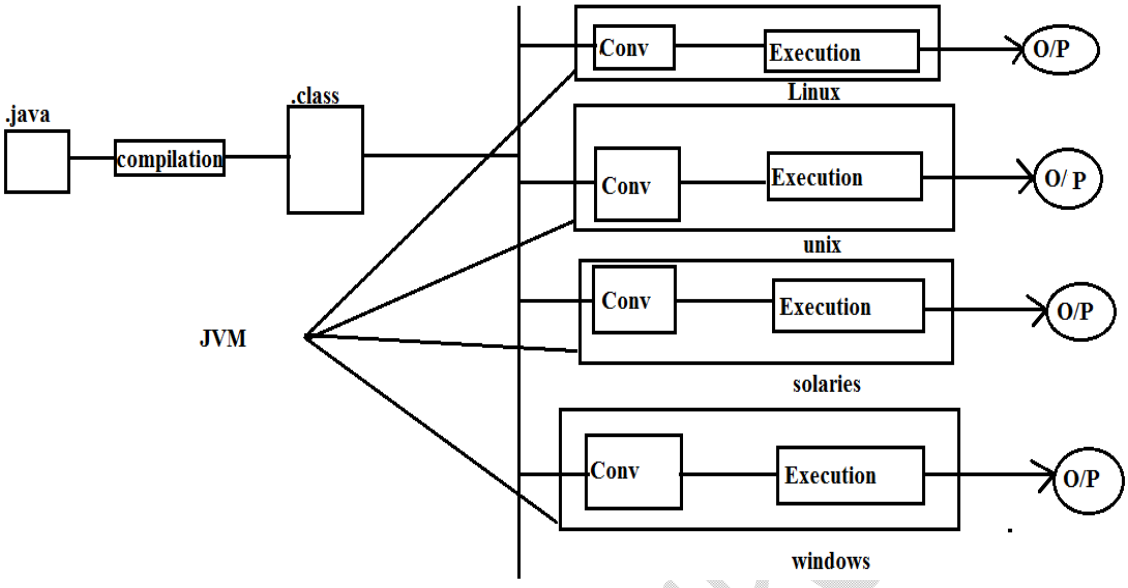
Ex: JAVA

If we compile Java program on windows operating system then compiler will generate .class file, it is neutral every operating system, which contains bytecode,it is an intermediate code, it is not directly executable code. If we want to execute neutral bytecode then we must require converters to convert neutral bytecode to the local operating system representations.

To perform the above conversions and to make Java as platform independent programming language,Sun MicroSystems has provided a tool called as "JVM"[Java Virtual Machine].

NOTE:Java is a platform independent programming language due to the availability of JVM.

NOTE:Java is a platform independent programming language but JVM is platform dependent tool.



Compilation of JAVA

4.Multiple inheritance is not possible in Java:

The process of getting variables and methods from one class to another class is called as Inheritance.

The main objective of inheritance is to provide "Code Reuseability".

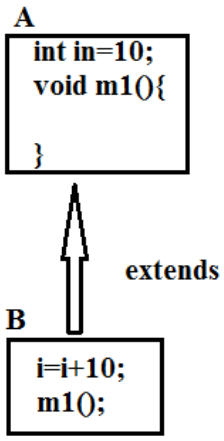
There are two types of inheritances at basic level of Object orientation.

1.Single Inheritance

2.Multiple Inheritance

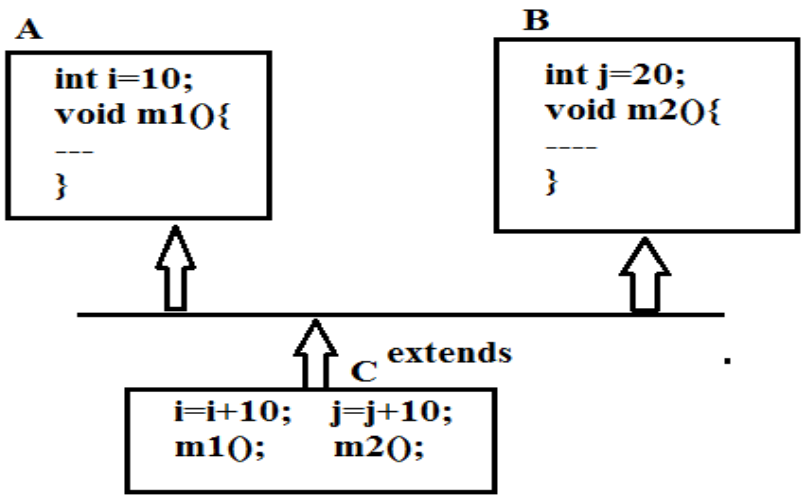
1.Single Inheritance:

The process of getting variables and methods from only one super class to one or more no.of subclasses is called as Single Inheritance.

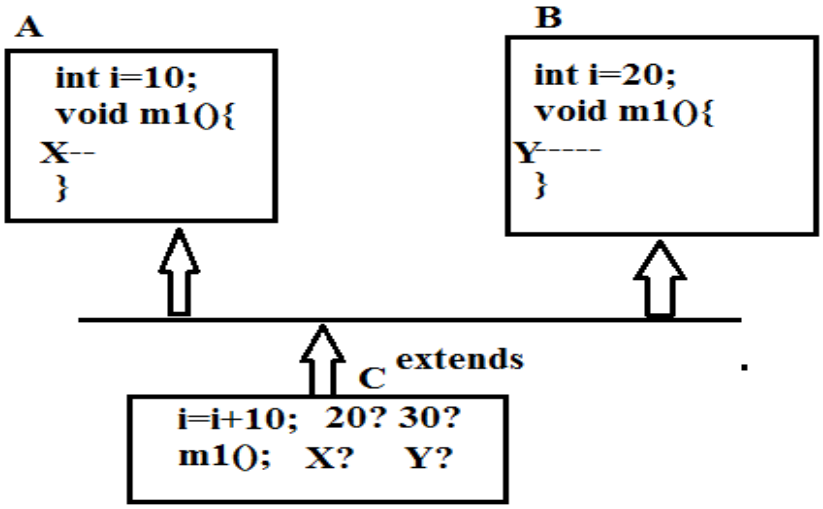


2.Multiple Inheritance:

The process of getting variables and methods from more than one super class to one or more no.of sub classes is called as Multiple Inheritance.



If we declare same variable with different data and same method with different implementation at both the super classes in multiple inheritance and if access that variable and that method in sub class then which super class variable will be accessed and which super class method will be accessed is a confusion state.



Java is a simple programming language,it should not allow confusion oriented feature,so that multiple inheritance is not possible in Java

5.pointers are available in C and C++ but pointers are not possible in JAVA:

pointer is a variable, it able to store the starting address location of a dataStructurer,where the dataStructurer may be a variable,an array or another pointer variable.

Pointers are not possible in JAVA, because

1.Pointer variables will be recognized and initialized at the time of compilation, pointer variables must require static memory allocation but JAVA is following dynamic memory allocation.

2.Pointer variables could be supported by static programming languages.

3.Pointer variables are suitable for platform dependent programming language but Java is platform independent programming language.

4.pointer variables will reduce security for the application data but Java is more secure programming language,it has to provide very good security for the application data.

5.Pointers will increase confusion to the developers but Java is a simple programming language, it should not provide confusion to the developers.

6.Destructors are required in C++ but Destructors are not required in JAVA:

In Object Oriented programming languages, it is convention to create objects at beginning in order to store data and it is convention to destory object at the end of our application.

In Object Oriented Programming language,to create objects we will use "constructor" and to destroy objects we will use destructors.

In case of C++,developers must take responsibility to create objects and to destroy object, with this, developers must require both constructors and destructors.

In case of Java, Developers should take responsibility only to create object, not to destroy objects will be destroyed automatically by an implicit tool called as "Garbage Collector".

Due to the availability of "Garbage Collector" in Java it is not required to use Destructors in Java.

Due to the unavailability of "Garbage Collector" in C++,it is required to use Destructors in C++

7.Operating overloading is not possible in Java:

Declaring one operator with multiple operations or functions is called as OperatorOverloading.

Operator overloading is not possible in Java,because,

1.Operator overloading is a rarely used feature in the application development.

2.Operator overloading is able to increase confusion to the developers.

NOTE:Java has declared some few predefined operators like +,*...as overloaded operators implicitly,but Java has not given any environment to do operator overloading explicitly at developers level.

Ex: int a=10;

int b=20;

```
int c=a+b;//  + operator is used to perform arithmetic addition.


String str1="abc";

String str2="def";

String str3=str1+str2;//here + is used for String concantation.
```

8.C,C++ are following "call by value" and "call by reference" parameter passing mechanism but Java is following only "call by value" parameter passing mechanism.

In any programming language,if we pass primitive data as parameters to the methods then the parameter passing mechanism will be call by value.

In any programming language,if we pass address location as parameter to the methods then the parameter passing mechanism will be "call by reference".

In the case of C and C++,if we pass pointer variable as parameter to the methods then the parameter passing mechanism will be "call be reference" because, pointer variable are able to store address location directly.

In the case of Java,if we pass reference variables as parameter to the methods then the parameter passing mechanism will be call by value only,because,in Java application reference variables are able to store object reference variable value,which is not address location.

9.In C and C++,2 bytes of memory is allocated for integer and 1 byte of memory is allocated for characters but in Java 4 bytes of memory is allocated for integers and 2 bytes of memory is allocated for character.

In general,in C and C++,memory allocation for the primitive dataTypes is not fixed,which is variable depending on type of operating system which we used.

In case of Java,memory allocation for the primitive dataTypes is fixed irrespective of the operating system which we used.In case of Java,4 bytes of memory is allocated for integers and 2 bytes of memory is allocated for characters.

Java Features:

To expose the nature of Java, Java has given the following features:

1.Simple

2.Object Oriented

3.platform Independent

4.Arch.Neutral

5.Portable

6.Robust

7.Secure

8.Dynamic

9.Distributed

10.Multi Threaded

11.Interpretive

12.HighPerformance

1.Simple:

Java is a simple Programming language,

Java has eliminated all the confusion oriented features like multiple inheritance,operator overloading,pointers...

Java applications will take less memory and less execution time.

Java is using all simplified syntaxes of C and C++.

2.Object-Oriented:

Java is Object-Oriented Programming language,because it able to store the data in the form of objects.

3.platform Independent:

Java is platform independent programming language,because it able to allow its applications to compile on one operating system and it will execute on another operating system.

4.Arch.Neutral:

Java is Arch.Neutral Programming language because it able to allow its applications, to compile one hardware arch.system and to execute on another hardware arch.system.

5.Portable:

Java is portable programming language because it able to allow its applications on any operating system and also to execute on any hardware system.

6.Robust:

If any programming languages is having very good at memory management system and also exception handling features then that programming language is called as Robust programming language.

Java is Robust programming language,because,

Java is having very good at memory management System that is HeapMemory,StackArea,Method Area.. and also memory management system will take only at runtime in Java.

java is having very good at exception handling mechanisms due to the availability of very good predefined library to represent each and every exceptional features.

7.Secure:

Java is a very good secure programming language,because,

Java is having a separate component like "securityManager" as part of JVM to provide implicity Security.

Java is having a separate middle ware service in the form of "JAAS"

[java Authentication and Authorization Service] to provide web Secuirty.

Java has provided implementation for all the network security algorithm to provide network security.

8.Distributed:

If we design any Java application with out using client-server arch. then that

application is called as "standalone application".

If we design any Java application using client-server arch. then that

application is called as "Distributed application".

9.Multi-Threaded:

Thread is a flow of execution to perform a particular task.

There are two types of thread Models:

1.Single Thread Model

2.Multi Thread Model.

Single Thread Model will allow only one thread at a time to perform tasks,it increase execution time,it decrease application performance.

Multi Thread model is able to allow to create and execute more then one thread at a time,

it will reduce application execution time,it will improve application performance.

Java is a multi threaded programming language,it will provide very good environment to design and execute more than one thread at a time.

11.Interpretive:

Java both compilative and inteprative programming language. because,

1)To check use provided syntax voilations we must require compilation for Java program.

2)To execute Java program,we must require an interpreter inside JVM.

12.HighPerformance:

Java is a high performance programming language due to its rich set of features like Robust,Dynamic,Secure,object oriented,platform Independent...

Java Programming Format:

To design java applications,we have to use the following Structure.

Comment Section	Optional
package Section	Optional
Import Section	Optional
Classes/interfaces Section	Optional
Main Class Section	Mandatory

Comment Section:

- As part of java applications development,it is convention to provide description about our application before starting application implementation,here the description may contain objective of implementation,Author details,Clients information,dead line details,module details...
- In the above context,to descrobe the above information we have to use Comment Sction,where we have to use Comments.

There are three types of comments in java:

1.Single Line Comment:

It will allow the description with in a single line.

Syntax:

//------description-------

2.Multi Line Comment:

It will allow the description in more than one line.

Syntax:

/*

---Description----

----

*/

3.Documentation Comment:

It will allow the description in more than one page.

Syntax:

/*

*---------------

----

*-------

*/

In general we will utilize documentation comment to prepare API documentation but it is not suggestable or recommended.

NOTE:API documentation is a document,which contains the description about classes declarations,constructors declarations,variables declarations,method declarations....

To simply API documentation,JAVA has provided a predefined tool in the form "javadoc".

C:\abc\Employee.java

```
/* Programme for preparing JAVADOC*/

public class Employee{

public static final String eid;

public String ename;

public float esal;

public String eaddr;

public Employee(String eid,String ename,float esal,String eaddr){}

public void add(String eid,String ename,float esal,String eaddr){}

public void search(String eid){}

public void delete(String eid){}

}
```

c:\abc>javadoc Employee.java

In java applications,to describe metadata JDK5.0 version has provided a new feature called as "Annotation".

Q)To describe metadata in java applications,already we are using comments then what is the requirement to go for "Annotations"?

Ans:In java applications,if we provide metadata by using comments then that metadata will be available upto .java file,it will not be available upto .class files and upto RUNTIME of our application,because,LexicalAnalyzer will remove the provided commented metadata from Java application as part of compliation.

As per the application requirement,if we want to bring the metadat upto .java file,upto .class file and upto RUNTIME of our application then we have to use "Annotation".

Q)In Java applications to provide metadata at runtime we are able to use XML documents but what is the requirement to go for annotations?

Ans:To provide metadata upto runtime of our java applications if we use XML documents then we are able to get following problems:

1)We must learn XML technology.

2)We have to check whether XML documents are located properly or not.

3)We have to check whether XML tags are formatted properly or not.

4)We have to check whether we are using right parsing mechanism or not to access the data from XML documents.

To Overcome all the above problems we have to use a java alternative that is "Annotations".

Example:

If we want to design any servlet by using XML document then we have to provide almost all 10 lines of code in XML file.

web.xml:

```
<web-app>

<servlet>

<servlet-name>ls</servlet-name>

<servlet-class>LoginServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>ls</servlet-name>

<url-pattern>/login</url-pattern>

</servlet-mapping>

</web-app>
```

LoginServlet.java

public class LoginServlet extends HttpServlet{

}

If we design any servlet by using annotations then it is sufficient to provide only one line of java code[Annotation]

@webServlet("/login")

public class LoginServlet extends HttpServlet{

}

NOTE:In JAVA/J2EE applications,we can utilize annotations as an alternative for XML tech or XML documents.

XML-Based Tech	Annotation Based Tech

JDK 1.4	JDK 5.0 and above
JDBC 3.0	JDBC 4.0
Servlets2.5	Servlets3.x
Struts1.x	Struts2.x
JSF1.x	JSF2.x
Hibernate3.2.2	Hibernate3.2.5
Spring2.x	Spring3.x
EJBs2.x	EJBS3.x

2.Package Section:

- Package is a collection of related classes and interfaces as single unit.

  In java,packages will provide the following advantages

  1.Modularity

  2.Abstraction

  3.Security

  4.Sharability

  5.Re usability

There are two types of packages in java:

1.Predefined packages

These packages are provided by java along with software.

Ex: java.io

java.util

java.awt

2.User-defined Package

- These packages are provided by the developers as per their application requirements.
- To declare user defined packages in java program we have to use "Package section".

If we want to declare user defined packages in java programme then we have to use the following Syntax.

package Package_Name;

where package declaration statement should be first statement

after comment section in java program.

Where package_Name must be unique,it should not be duplicated and it should not be sharable.

To provide package names,Java has a given convention like to include our company domain name in reverse.

Ex:

package com.durgasoft.icici.transaction;

com.durgasoft---->company domain name in reverse

icici--------->client name/project name

transaction---->module name

Q) Is it possible to provide more than one package declaration statements with in a single java file?

Ans:No,it is not possible to provide more than one package declaration statements with in a single java file,because,always package declaration statement must be first statement in java file

Ex:

abc.java

package p1;----valid

package p2;--->Invalid

package p3;--->Invalid

import Section:

- In general,we are able to design java applications by distributing all the classes and interfaces over multiple no of packages.
- In the above context,if we want to use a particular package classes and interfaces in the present java file,first,we have to make available that package provided classes and interfaces to present java file.
- To make available a particular package provided classes and interfaces to the present java file,we have to import the respective package to present java file

To import a particular package into present java file,we have to use the following syntaxes.

import package_Name.*;

It will import all the classes and intefaces from the specified package.

Ex:

import java.io.*;

import package_Name.Member_Name;

It will import only the specified class or interface from the specified package.

Ex:

import java.io.BufferedReader

Q)Is it mandatory to import packages to use classes and interfaces of a particular package in present java file?

Ans: No,It is not mandatory to import packages in java file inorder to use the classes and interfaces of a particular package,because it is possible to use classes by using their fully qualified names without using "import" statements.

NOTE:Specifying classes along with their package names is called as Fully qualified name.

Ex:java.io.BufferedReader,java.util.ArrayList

java program with import statement:

```
import java.io.*;

BufferedReader br=new BufferefReader(new InputStreamReader(System.in));
```

java program with out import statement:

```
java.io.BufferedReader br=new java.io.BufferedReader(new
java.io.InputStreamReader(System.in));
```

NOTE:In java files,we are able to provide at most one package declaration statement but we are able to provide more than one import statement.

classes/interfaces section:

- In java applications,the main purpose of classes and interfaces is to represent all real world entities like Persons,Animals,Students,Customers,Account....
- In java program,to write classes and interfaces we have to use classes/interfaces section.
- In java files,we are not having any restrictions about the no classes and interfaces,it is possible to provide any no of classes and interfaces with in a single java file but it is suggestable to provide one class or one interface per java file.

Note:In java,a single .java file may contains more than one class but a single .class file contains only one class or one interface.

Main class Section:

The purpose main() method in java application is,

1.To manage application logic in java applications.

2.To define starting point and ending point for the application execution.

Note:In general,main() method starting point is the starting point of the application execution and main() method ending point is the ending point of the application execution.

- In java applications,we will provide main() method in a class called as "Main Class",it is mandatory for each and every java application beacuse main() method is mandatory to start application execution.


LEARN FROM EXPERTS ...
COMPLETE JAVA
CORE JAVA, ADV. JAVA, ORACLE, STRUTS, HIBERNATE, SPRING, WEB SERVICES,...
COMPLETE .NET
C#.NET, ASP.NET, SQL SERVER, MVC 5 & WCF
TESTING TOOLS
MANUAL + SELENIUM
ORACLE
D2K
MSBI
SHARE POINT
HADOOP
ANDROID
C, C++, DS, UNIX
CRT & APTITUDE TRAINING
AN ISO 9001:2008 CERTIFIED
DURGA
Software Solutions®
202, 2nd Floor, HUDA Maitrivanam,
Ameerpet, Hyd. Ph: 040-64512786,
9246212143, 8096969696
www.durgasoft.com