**Java Basics**

* Define the scope of variables.
* Define the structure of a Java class
* Create executable Java applications with a main method; run a Java program from the command line; produce console output
* Import other Java packages to make them accessible in your code
* Compare and contrast the features and components of Java such as: platform independence, object orientation, encapsulation, etc.

.java (platform independent) 🡪 compile (platform independent) 🡪 bytecode (platform independent) 🡪 jvm 🡪 machine code (platform dependent)

JDK – JRE + development tool

JRE – JVM + library classes

JVM –

Int a = 10 ; a – variable, 10 – literal

**A screenshot of a computer

Description automatically generated**

By default Integral value is int

Decimal - 0 to 9 , int x = 10;

Octal - 0 to 7, must start with 0, int x = 010

Hexadecimal - 0 to 9, a to f - 0X10; Note: case sensitivity does not matter in hexadecimal value

Note: there is no way to assign explicitly a byte type or short type.

Byte b = 100b; - invalid , Short s = 100s; - invalid

To Decimal we can assign each type of value – octal, hexadecimal, decimal.

By default, floating value is double

Note: floating point literal only accept decimal , hexadecimal and octal can’t be assigned to the floating point literals.

In other case, integral literal, octal decimal and hexa can be assigned to integral literals.

Exponential form – it is always double

Double d = 1.2e3;  1.2\*10^3  1.2\*1000 1200.0

Byte 🡪 short 🡪

Int🡪 long🡪 float🡪 double

Char🡪

Int [4] x; - not allowed at the time of declaration we are not allowed to specify the size.

Note: if you want to specify dimension before variable this facility is only available for the first variable

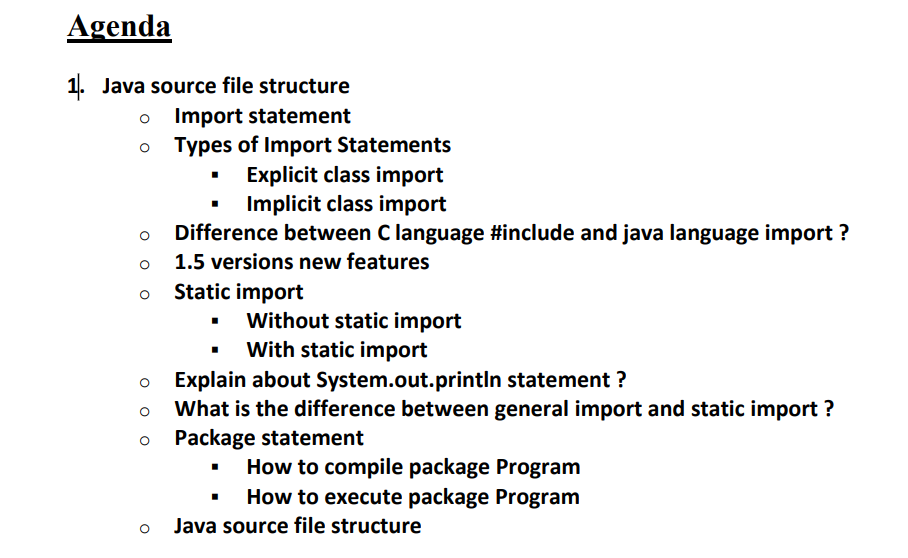
Int [] x = new int [3];

Sop(x); // [[1@hashcode

Int []x;

X = {10,20,30} // invalid – all the decleration should be in single line for shortcut method

8. Declaration and Access Modifiers





**Done all**

Note: - 1. Program contain any number of class

2. If none of the class is public then source file can be anything

3. But, if class contain public class then source file name must be same as public

4. one source file can contain atmost one public class

5. If both B and C classes are declared as public and name of the file is B.java then we will get compile time error saying "class C is public, should be declared in a file named C.java".

6. It is highly recommended to take only one class for source file and name of the Program (file) must be same as class name. This approach improves readability and understandability of the code.

7. for each java class one .class file will be created on compile using javac <filename>

8. if the class has main method then it will run otherwise " "main" java.lang.NoSuchMethodError: main

**Import statement:**

* 1. Explicit
  2. Implicit

If two import packages have same class then it give **error: reference is ambiguous**

**D:\Java>javac Test.java Test.java:7: reference to Date is ambiguous; both class java.sql.Date in java.sql and class java.util.Date in java.util match**

While resolving class names compiler will always gives the importance in the following order.

1. Explicit class import

2. Classes present in current working directory.

3. Implicit class import.

**Whenever we are importing a package all classes and interfaces present in that package are by default available but not sub package classes.**

In any java Program the following 2 packages are not require to import because these are available by default to every Java Program.

1. java.lang package

2. default package (current working directory)

"Import statement is totally compile time concept" if more no of imports are there then more will be the compile time but there is "no change in execution time".

import static java.lang.Math.sqrt; if we have so many static variable to use .. so using class name every time is not feasible

While resolving static members compiler will give the precedence in the following order.

1. Current class static members

2. Explicit static import

3. Implicit static import.

**NOTE:** for normal import we import till class - **import java.lang.Math;**

But for static import we import till method and variable - **import static java.lang.Math.sqrt;**

package com.durga;

class Test

{

public static void main(String[] args)

{

System.out.print("first class");

}

}

To run it via command line if package is available

* Javac Test.java
* Java com.durga.Test **// java Test (if package is not available)**

2 package statement is not allowed.

In any java Program the 1st non comment statement should be package statement [if it is available] otherwise we will get compile time error.

**Order of programme:**

Package – single only

Import- any number of

Class/interface/enum/decleration – any number of

**\*\*\* what valid java program can contain**

* + - Only package - valid
    - Only import - valid
    - Only package and import - valid
    - Only class - valid
    - Both package , import and class - valid