First Java Program & Main method in Java

classmate Date 12/10/2022 Page

⇒ Book:- Orocle's Java the complete reference
⇒ IT:- Information Technology → it's all about data.
⇒ Jshell:- Jshell is introduced shell. This tools allows you to execute java code, getting immediate results.
ex:- 2+3 output: $1 ==> 5 (→ variable, → ans)
→ jshell > / → show all the command by /
→ jshell > /help → list of all features/comments in jshell.
⇒ IDE:- (Integrated Development Environment) where we can type, compile, Run, De-bug, Test our code.
ex:- Eclipse(we use this), intellij idea, VS code.
⇒ Editors:- where we can only type our code.
sublime editor, notepad, notepad++, etc.....
Q) Can we run a Java file without compilation?
⇒ yes (from JDK 11) we can run directly without javac command. all the compilation happened background automatically. (Not recomm)
Q) What happened when we created FirstCode.java → filename SecondCode → give classname and run it?
⇒ Then, if we compile javac FirstCode.java it will run and create a class name of SecondCode.class Because we gave different classname. (SecondCode)
now, if we run as normal java FirstCode it will not run, we have to do java SecondCode to successfull run.
→ If we create two class like that it create also 02 file of class.
⊛ .jar ⇒ Java archiver (The bundle of .class files)
→ on old Nokia mobile we had to download .jar file to play a game.

⇒ First Program of JAVA →

```
import java.util.*;          → import statements
package com.sample;          → package statements
public class Main{
(Access specifier) (naming conv.) (name)
    public static void main(String[] args){      → parameter (string type arguments) → variable
    (Access specifier) (keyword) (Return type) (name)
        System.out.println("Hello world!");
        (final class)  variable  method of PrintStream class
    }
}
```

-Output: Hello world!

classmate
Date
Page

⇒ APP/software/program ⇒ same ✓

I.Q) ⇒ What is main method in java?

→ Starting point has to defines in java and that is (main)

OS → operating system (boss of our computer)

$$\begin{tikzcd}
\boxed{\text{our code}} \arrow[r] & \text{Javac} \arrow[r] & \text{.class} \arrow[r] & \text{JVM} \arrow[l, bend left] \arrow[d] \\
 & & \text{main} & \text{jvm search for main method to start} \arrow[l]
\end{tikzcd}$$

Ans:- ① What ever task or work or activity we should write under method. and the starting point of a code is main method (must have)

② Java → method

⇒ (i) name (ii) parameter, (iii) Body, (iv) return type

Syntax:-

```
Accessmodifier, return type, methodname, (parameter) {
                                           ↗ inputes
{

    task/body/code
    return ;

}
```

(iv) Return types:-

```
(void) display( ){            (int) display()
    S.O.P("Hello");            {
                                  S.O.P(=);
                                  return 10;
    display();                 }
                                  display();
```

when we don't want any return from the method

if we want something in return write the type of datatype.

outsider
```
class Demo(){
  m1();
  m2();
}
```

⇒ We can't use any method outside of any class without creating object of that class.
But you can use there is a way that is done by (Static) keyword.

→ main → name & JVM will search for this name only.

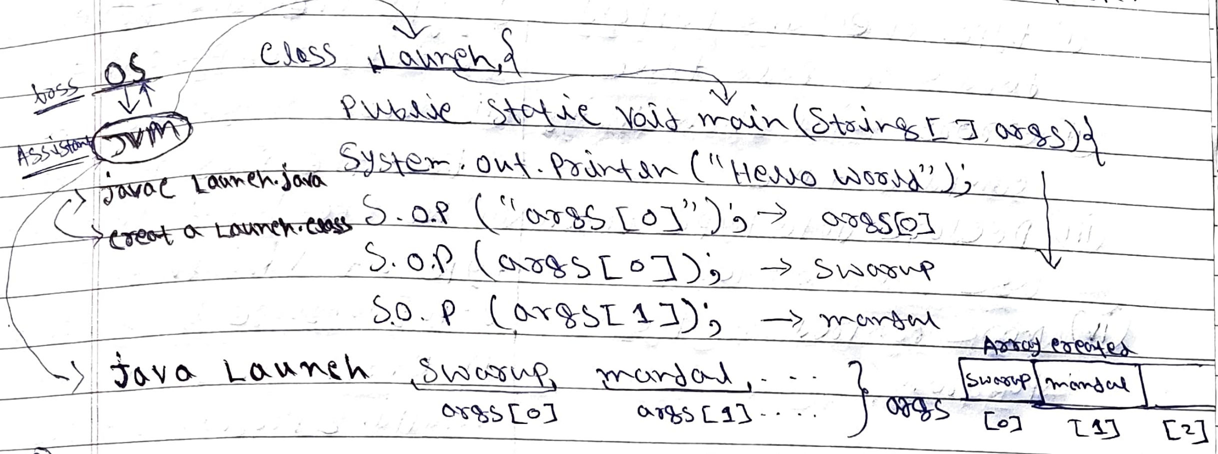
→ void → it is a return type (Java main() will not anything). But you also creat as int, float, double, String etc

→ Public → Access specifier (To increase visibility)

→ Static → (keyword) can be allowed without object creating.

→ String[] args → to recive command line arguments.
↳ we can give any name.

Q) why need of commandline args?
=> To pass the data from command line to a method during Execution.
→ args => (arguments) means you are giving information.



```
class Launch{
    Public static void main(String[] args){
    System.out.println("Hello world");
    S.o.p ("args[0]"); → args[0]
    S.o.p (args[0]); → Swarup
    S.o.p (args[1]); → mandal
```

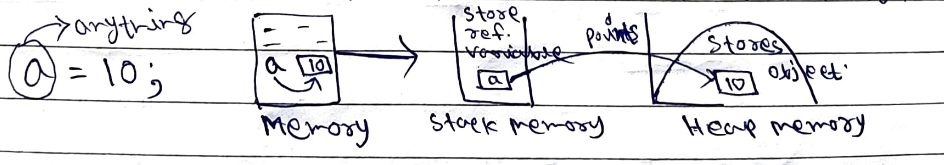
(★) => How to run commandline Arguments in eclipse?
right click on you program files > Run as > Run configure > select arguments > give arguments on program arguments > Run

(★) Valid Syntax:-

Public static void main (String[] args) ✓
Public static void main (String args[]) ✓
Static Public void main (———) ✓
Static Public main void (———) ✗
Public Static void main (String... args) ✓
Public Static void main (int [] args) ✗

(★) Variable (★)

→ Variable is a container where we can store different kind of data. Specify that data by datatype.



I.Q) What is Statically typed & Dynamically typed Programming Languages? What is the difference between them.

(S.T) (D.T)

(in c, c++ Java) a = 10;
S.o.p(a);
output => give error

(in python Javascript) a = 10;
Print (a);
output :-> 10 (give error)

⇒ Static Language (VS) Dynamic Language

Static Language	Dynamic Language
(i) type checking happend during compile time	(i) type checking happened during ~~compi~~ Run time
(ii) errors will show during compile time	(ii) errors might not show till the program runs.
(iii) Declare datatype before use [int a=10;]	(iii) No need to declare data-type of variables a=10[language by itself identify]
(iv) more control over the program	(iv) ~~error~~ saves time in writting program but error might give at runtime.

Q> What is the purpose of giving arguments at the time of execution? (During spring Boot we will understand very clearly)

⇒ ~~Developme~~ code :- 3 Phases before relasing the code to users.

(i) Development :- few inputs $\overset{\rightarrow \text{जुड़}}{\text{associated}}$ with developers.
(ii) Testing:- few inputs associcted with test engineers.
(iii) Production :- few inputs associated with QA.

⇒ JDK ⇒ compiler + JRE (jvm (JIT compiler) + library tools)

Note:- JDK is required for Developers (to write, run code)
JRE is required for Endusers, (to only run code)

⇒ Static Language (VS) Dynamic Language

Static Language	Dynamic Language
(i) type checking happend during compile time	(i) type checking happened during ~~compi~~ Run time
(ii) errors will show during compile time	(ii) errors might not show till the program runs.
(iii) Declare datatype before use [int a=10;]	(iii) No need to declare data-type of variables a=10 [language by itself identify]
(iv) more control over the program	(iv) ~~error~~ Saves time in writting program but error might give at runtime.

Q> What is the purpose of giving arguments at the time of execution? (During Spring Boot we will understand very clearly)

⇒ ~~Developme~~ code:- 3 Phases before releasing the code to users.

(i) Development:- few inputs associated (→ [illegible]) with developers.
(ii) Testing:- few inputs associated with test engineers.
(iii) Production:- few inputs associated with QA.

⇒ JDK ⇒ compiler + JRE (jvm (JIT compiler) + library tools)

Note:- JDK is required for Developers (to write, run code)
JRE is required for Endusers, (to only run code)