Agenda

- OOP Revision
- History
- Java Versions
- Java Platforms
- Installation
- Object Oriented
    - class
    - object
- Command line compilation(Hello World)
    - Helloworld compilation,execution and explaination
- Hello World in STS
- compilation and execution with src and bin directory
- CLASSPATH
- JRE/JDK/JVM
- multiple main method
- Java Langugage Specification for public class.
- Console input and output
- BuzzWords

Java Module

- 100 Marks
    - 40 Marks for Lab Exam
    - 40 Marks for CCEE
    - 10 Marks for Assignmnets
    - 10 Marks Quiz / CaseStudy
- 18 Days
    - Java 1.1
        - 6 days
    - Java 1.2

OOP

- Major Pillars
    - Abstraction
    - Encapsulation
    - Modularity
    - Hirerachy
- Minor Pillars
    - Polymorphism/Typing
    - Concurrency
    - Persistance

History

- In 1991, the team from sun microsystems led by James Gosling and Patrick naughtan decided to cretae their language which can work on smaller devices like remote control, or cable tv boxex.
- this team was called as Green team
- James Gosling had the vision that the language should be capable of running on devices with low memory
- Because these developers were from UNIX background they decided to keep their base language as C++
- Within an year they came up with their first product called as '*7' (Smart Remote Control)
- Nor the sunmicrosystem , nor the consumer electornic companies were intrested in this product.
- So the team deciced to come up with some better ideas to create a new product, and to market it in some better way.
- James Gosling decided to name the language as OAK, however the language with OAK already existed it was further changed to JAVA
- Meanwhile World wide web(WWW) was getting popular, and the key for this was a browser that can translate the hyper text pages to the GUI/Screen
- The team cam up with a browser called as HOT Browser which was capabale of running the java code inside it.
- The browser was dynamic i.e it can work into real time tranfering the data back and forth.
- this java code that was able to run inside the brower was called as Appletes

Java Versions

- JDK Beta - 1995
- Java 1.0 - 1996
- Java 1.1 - 1997
- J2SE 1.2 - 1998
    - Collections
- J2SE 5 - 2004
    - Annotations
    - Generics
    - Enum
- Java SE 8 - 2014 (LTS)
    - Functional Programmming
    - Lamda Expressions
- Java SE 11 - 2018 (LTS)
- Java SE 17 - 2021

Java Platforms

- 1. Java Card
    - Used to develop applications on very small devices like smart cards
- 2. Java ME (Micro Edition)
    - Is used to develop applications for small mobile devices which are low in memory
- 3. Java SE(Standard Edition)
    - Used to develop desktop applications

- 4. Java EE (Enterprise Edition)
  - used to deveop web applications

Installation

- Follow the steps given in the installation.txt file
- once installation is done test the java version and STS
- For documentation
  - Java 8
    - https://docs.oracle.com/javase/8/docs/api/index.html
  - Java 11
    - https://docs.oracle.com/javase/11/docs/api/index.html

class

- It is a logical entity
- It conists of
  - Fields(Data Members)
  - Methods (Member Functions)
    - static method
      - that are designed to call on classname using . operator
    - non static method
      - that are designed to call on class objects using . operator

object

- It is a physical entity
- It is also called an instance of the class

Command line compilation(Hello World)

- For Compilation
  - javac <name of .java file>
- For Execution
  - java <name of .class file>

Understanding of main()

- in java the main method is defined as
  - public static void main(String args[])
- the main method is invoked by the JVM.

- It calls this main method directly on the classname without cretaing the object that is why it is made as static
- Main method does not return anything towards the JVM , thats why its return type void
- The main method should be accessiable outside the class that why it is made as public
- the main method tales the commad line arguments and hence it has an array of String as an parameter

Understanding System.out.println()

- System is a class declared inside java.lang package
- out is a static field declared inside System class
- out is an object of PrintStream class.
- println() is a method declared inside PrintStream class

STS Steps

- Change the workspace every day (Choose daywise workspace)
- Once STS is launched change the perspective to Java
- click on File -> new -> Java Project
- Select the Java version as Java SE 1.8
- click on finish.
- Right click in src -> new -> class
- Provide the class name, if main method is required select it and click on finish
- to execute click on run button

compilation and execution with src and bin directory (Demo01)

- Create directory Demo01
- Crete 2 sub directories src and bin
- Inside src cretae a Program.java file
- For compilation and execution use the below commands

```
// from the src directory open the terminal
javac -d ../bin Program.java

// set the CLASSPATH
export CLASSPATH=../bin

// execute the code
java Program
```

PATH

- It is an Operating system variable
- Used to keep the path of executable files.

CLASSPATH

- It is a java variable used to set the paths of all the .class file.
- We can set multiple CLASSPATH seperated by : in linux and ; in windows
- In linux to set the CLASSPATH
    - export CLASSPATH=<path of the .class file till bin directory>

JRE/JDK/JVM

- SDK(Software Development Kit)
    - tools + libraries + docs + Runtime Environmnet + IDE
- JDK (java Developemnt Kit)
    - tools
        - javac
        - java
        - javap
        - jar
    - docs
        - manuals or docs for using the tools and libraries
    - libraries
        - core libraries
    - JRE (Java Runtime Environment)
        - rt.jar (till Java 8)
        - JVM (Java Virtual Machine)

To create STS Shortcut on Desktop

- copy the sts.desktop file on your desktop
- change the path of Exec and icon to your STS executable path and the icon path
- save the file.
- Rightclick and allow it for launching

main method variations

- In java we can have multiple main inside single java project.
- Every class can have a main method
- Definining multiple main with same signature inside same class is not allowed
- If we change the return type of main jvm throws an error Main method must return a value of type void in class
- If we make the main method as non static jvm throws error Main method is not static in class
- If we remove the public access modifier of the main then jvm throws error Main method not found in class
- If we dont pass the String[] args as paramaneter to the main or pass any differnt type of parameter then jvm throws Main method not found in class
- If we make the main method name in caps then jvm throws error Main method not found in class
- main method overloading is allowed

Q.Why name of public class and java file name is same

- It is the java language specification to define the public classes in its own .java file

Q.Can we define multiple public classes in single .java file

- No We cannot

Q. Why to make class as public

- To maintain the visibility of the classes outside the package or in the different packages the classes need to be public
-

Console Input and Output

- Theor are two ways to perform input and output in java
    1. using Scanner class
        - It is present inside java.util package
        - to create the object of scanner class use below syntax

```
Scanner sc = new Scanner(System.in);
```

    2. Using Console class
        - It is present inside java.io package
        - to create the object of Console class use below syntax

```
Console console = System.console();
```

        - To execute the code where Console class object is cretaed we need to execute it through the terminal.
        - execution in STS will cause NULLPointerException

Java BuzzWords

- 1. Simple
    - Java was simple till Java 1.2
    - From java 1.2 onwards many classes were added which made java very powerful but made it too complex
    - It is simple for professional programmers
    - java have removed the rarely used concept of operator overloading
    - Java have removed the Complexity of Pointers
- 2. OOP
    - Java is an OOP Language
    - it supports all the major as well as minor pillars of OOP
- 3. Compiled and Interpreted
    - Java is both compiled as well as interpreted language

- 4. Architecture Netural
  - It follows WORA - Write Once Run AnyWhere
  - we can execute the comiled java code (.class) on any architecture
- 5. Portable
  - java is Portable because of the JVM
- 6. Distributed
  - Java Applications can be distributed on the network where multiple developers can work on the single project
  - Accesing the java objects on such distributed networks is same as that of accessing it on local machine
- 7. Robust
  - Java is robust because of its automatic memory management.
  - It is carried out with the help of Garbage Collector
- 8. MultiThreaded
  - Java suports multithreading
  - When we execute the java application two threads are started
    - 1. main Thread
    - 2. Garbage Collector Thread
      - Works in the background
- 9. Secure
  - you cannot access the physical memory directly of the maachine you are working on.
  - We deal directly with the virtual memory from the JVM
- 10. Dynamic
  - It supports Runtime type information which helps java to identity objects dynamically at runtime
- 11. High Performance
  - It is beacuse of the JIT Compiler
  - When a method is called multiple times then JIT compiler compiles the code in native form and stores it into the cache
  - So when such methods are called, jvm does not interpret them but uses the native code directly provoided by the JIT compiler.

LabWork

- class
  - class members
- object
- pointer -> dynamic objects
- reference