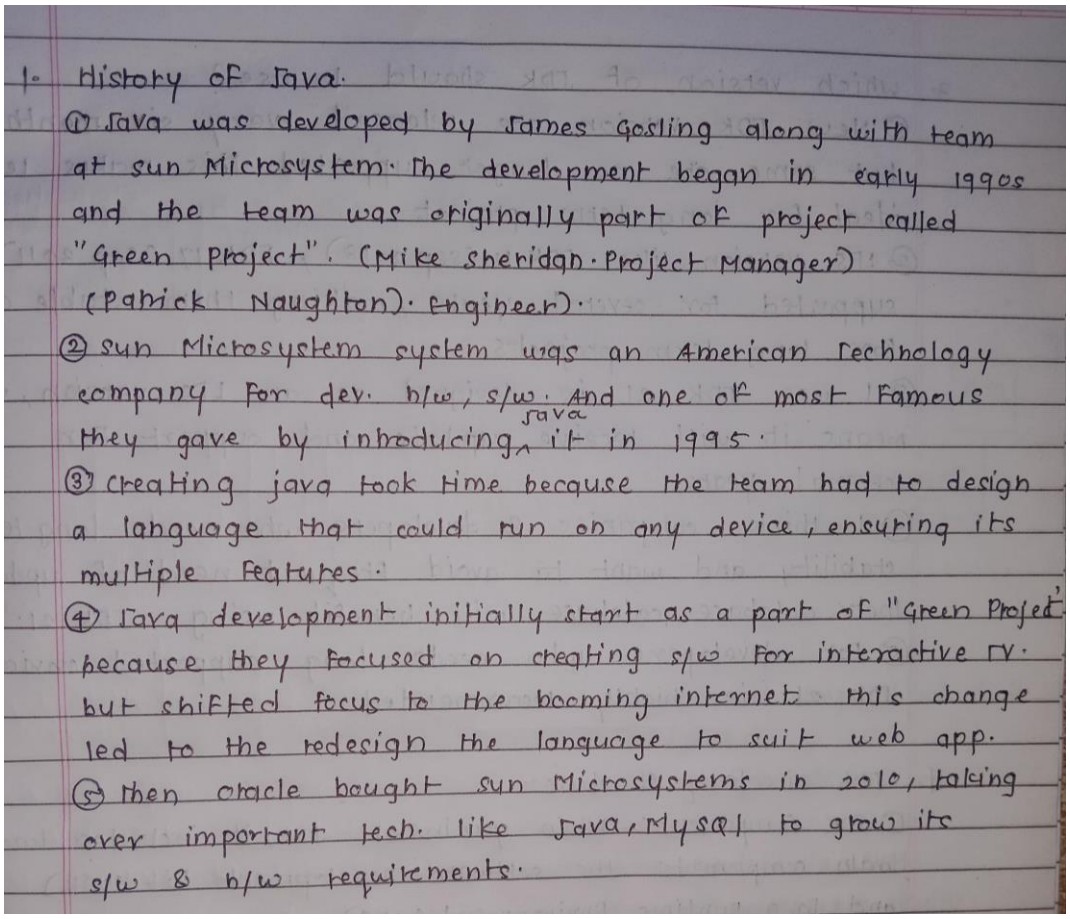
Assignment 1

Solution :

1. Reading Assignment: A Short History of Java



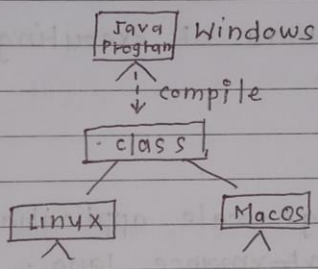
1. History of Java.
① Java was developed by James Gosling along with team at sun Microsystem. The development began in early 1990s and the team was originally part of project called "Green project". (Mike Sheridan. Project Manager) (Patrick Naughton). Engineer).
② Sun Microsystem system was an American technology company for dev. h/w, s/w. And one of most famous they gave by introducing, Java it in 1995.
③ Creating java took time because the team had to design a language that could run on any device, ensuring its multiple features.
④ Java development initially start as a part of "Green Project" because they focused on creating s/w for interactive TV. but shifted focus to the booming internet. This change led to the redesign the language to suit web app.
⑤ Then oracle bought sun Microsystems in 2010, taking over important tech. like Java, MySQL to grow its s/w & h/w requirements.

2. Reading Assignment: Java Language Features

Features of Java -

1. Platform Independent - (Write Once Run Anywhere).
- ~~We can compile any Java file on any one Platform~~
- Java file can be compiled on any platform and can be executed on other platform also.
- ~~If we create Java~~

2. Portable -



3. secured -
- JVM plays vital role to provide security.
- It verifies byte code.
- The JVM guarantees that there is no unsafe operation going to execute.
- Automatic memory management is done in Java refers to process of allocating and freeing up space for objects.

4. object oriented -
- It is based on the concept of objects and classes.
- Java supports concepts of OOPS - Inheritance, Data Abstraction, Polymorphism and data enscapsula tion.

5. Robust -
- Exception Handling features. Exception is an error which occurs at runtime. If exception occurs while running program then program gets terminated to avoid this problem. Java supports exception handling.
- Memory management features, most C and C++ programs crash in middle because of not allocate memory. ~~in~~ Such problem will not occur in Java because memory management is done by JVM.

6. Multithreading -
- in Java is a process of executing multiple threads simultaneously.

7. Scalable -
- It can handle large-scale applications with ease.
Java is high performance lang.

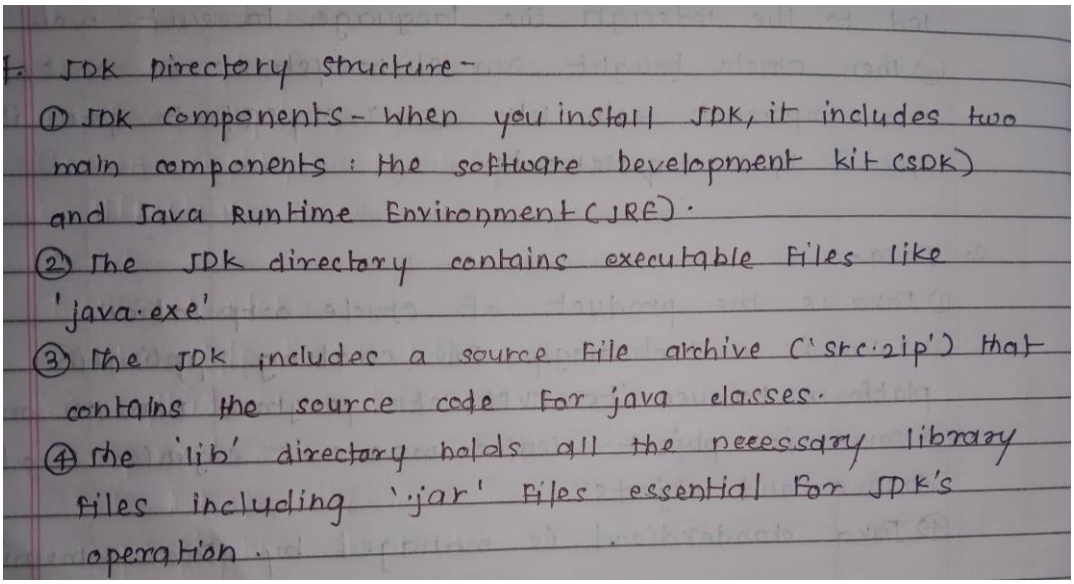
8. Standard libraries / support -
- Java contains list of libraries to make work easier.
- These libraries are called at run time by JVM.

9. Java is simple -
- Java is very easy to learn, and syntax is simple, clean and easy to understand.

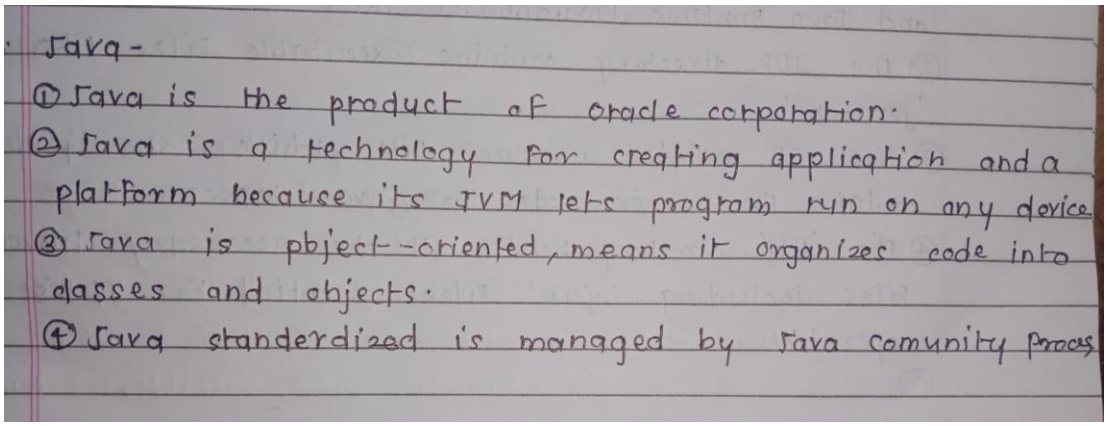
3. Reading Assignment: Which Version of JDK Should I Use?

Which version of JDK should I use?
① New JDK version are released every six months, but some are no longer supported. this makes less ideal for long term project.
② LTS version like JDK 11 (Sep 2018), JDK 17 (Sep 2011) are supported for several years, making them stable choice for long term projects.
③ I use JDK 21 is designed as an LTS version, which means it will receive update and support for several years.
④ I think companies & developers who needs long-term stability and want to avoid frequent need of update their database, codebase will benefit using JDK 21.
⑤ For developers who needs long supported environmen JDK 21 is highly recommended.

4. Reading Assignment: JDK Installation Directory Structure


JDK directory structure -
① JDK components - When you install JDK, it includes two main components : the software development kit (SDK) and Java Runtime Environment (JRE).
② The JDK directory contains executable files like 'java.exe'
③ The JDK includes a source file archive ('src.zip') that contains the source code for java classes.
④ The 'lib' directory holds all the neeessary library files including '.jar' files essential for JDK's operation.

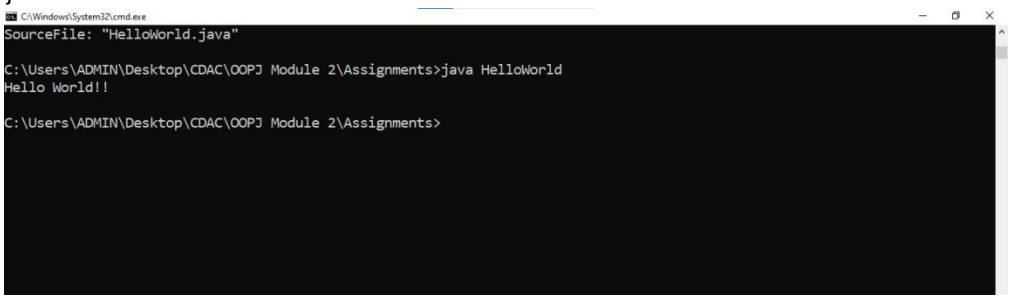
5. Reading Assignment: About Java Technology


Java -
① Java is the product of Oracle corporation.
② Java is a technology for creating application and a platform because its JVM lets program run on any device
③ Java is object-oriented, means it organizes code into classes and objects.
④ Java standerdized is managed by Java comunity Proces

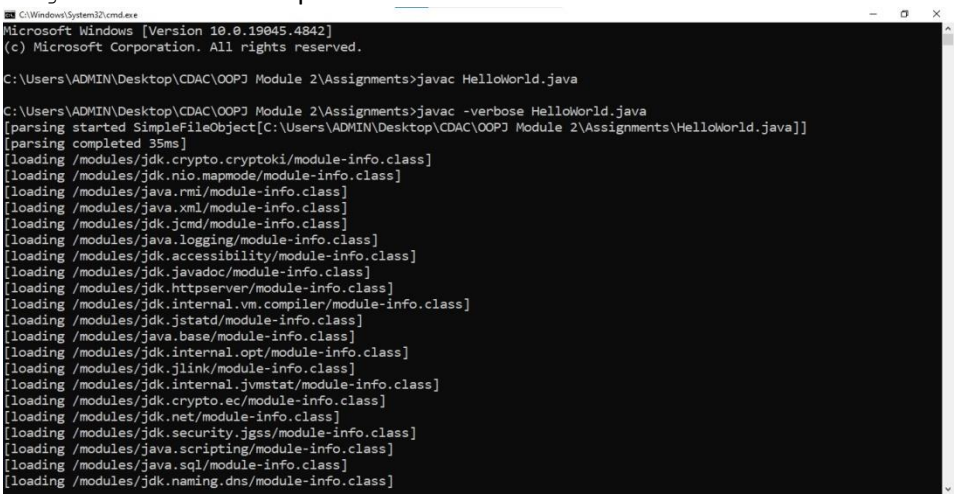
6. Coding Assignments

1. **Hello World Program**: Write a Java program that prints "Hello World!!" to the console.

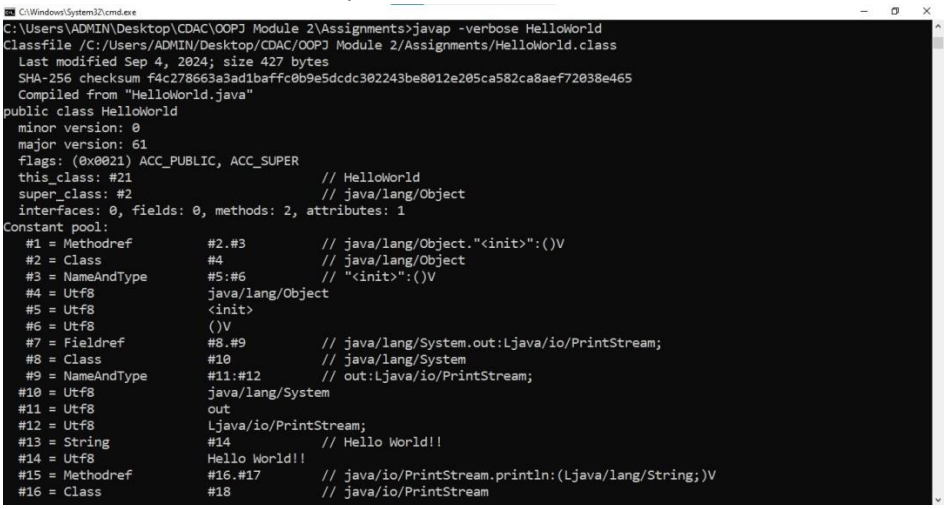
```
public class HelloWorld{
        public static void main(String[] args){

                System.out.println("Hello World!!");
        }
}
```



2. **Compile with Verbose Option**: Compile your Java file using the `-verbose` option with `javac`. Check the output.



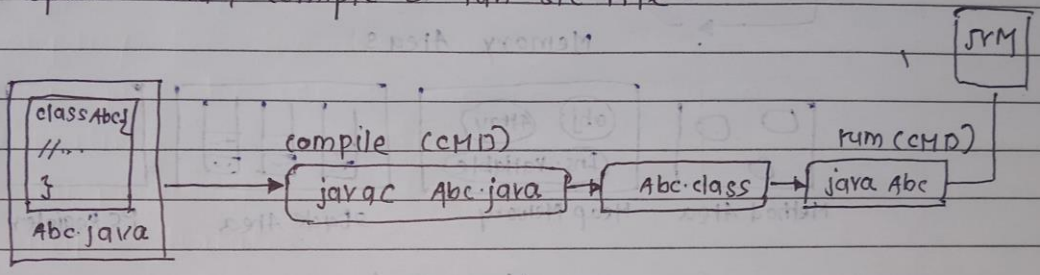
3. **Inspect Bytecode**: Use the `javap` tool to examine the bytecode of the compiled `.class` file. Observe the output.



7. Reading Assignment: The JVM Architecture Explained

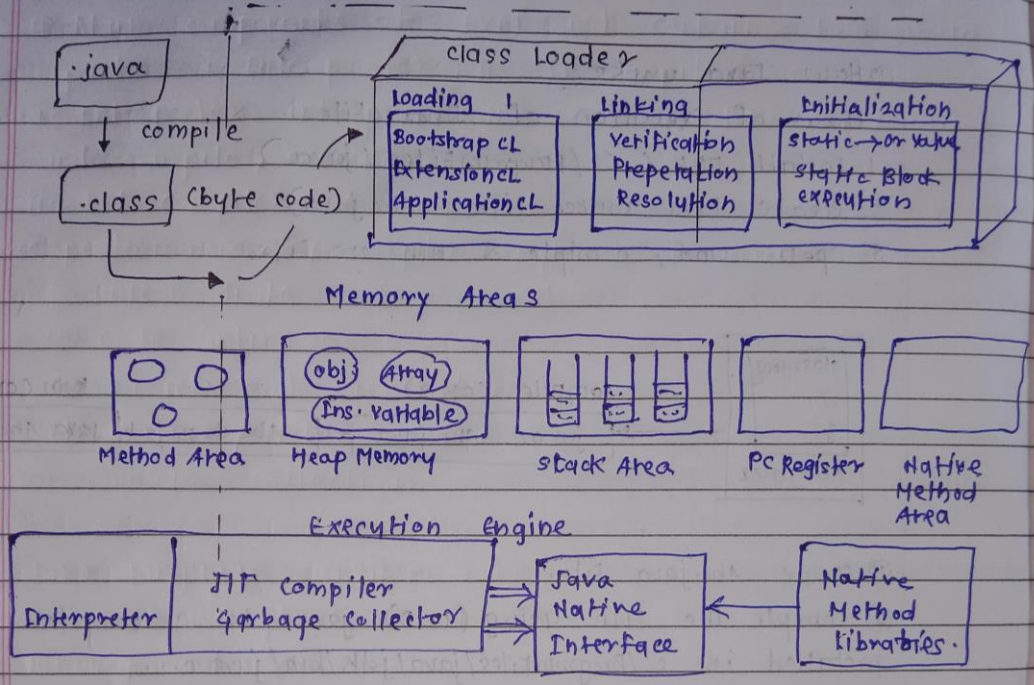
7. JVM Architecture –

① How Java works?
- How of Execution of Java File.
1. Install JDK (C:/ProgramFiles/java)
2. Create Java source file. (Abc.java)
3. Open cmd, compile & run src file.



① Create Abc.java file.
② Compile the file using (CMD). javac is a compiler which is installed in c:/ProgramFiles/java/jdk/bin/javac...
③ After compile the .class file will be generated.
④ Then run the file
⑤ After running the class file will be loaded to JVM.
⑥ Java virtual Machine. A s/w who not exist physically.
loads the .class file
⑦ JVM contains Interpreter and JUST In Time compiler they works (Interpreter + JIT compiler) together to execute .class file
⑧ Then finally ~~out~~display the ouput on CMD.

② Internal Working of JVM-



① After compiling .java gets converted into .class file also known as byte code. Then it is loaded to JVM.
② There are three main components of JVM.

1. Class Loader-
① .class file is loaded to memory using class loader.
② To load .class file to memory there are three components.
i. Bootstrap, ii. Extention, iii. Application class loader.
③ Using this it loads file to the Method area of Memory Method Area (class level data, class name, variable, constructor information.

④ JVM creates the object of class file in heap memory. Arrays are also stored in heap area. / Ins. variable.
⑤ Class loader is also responsible for linking. + where three process are carried out - i. Verification, ii. Preperation, iii. Resolution.
i. Verifies the byte code.
ii. Allocates the class level data into memory and initialize default value.
iii. Replacing symbolic names with originally name references.

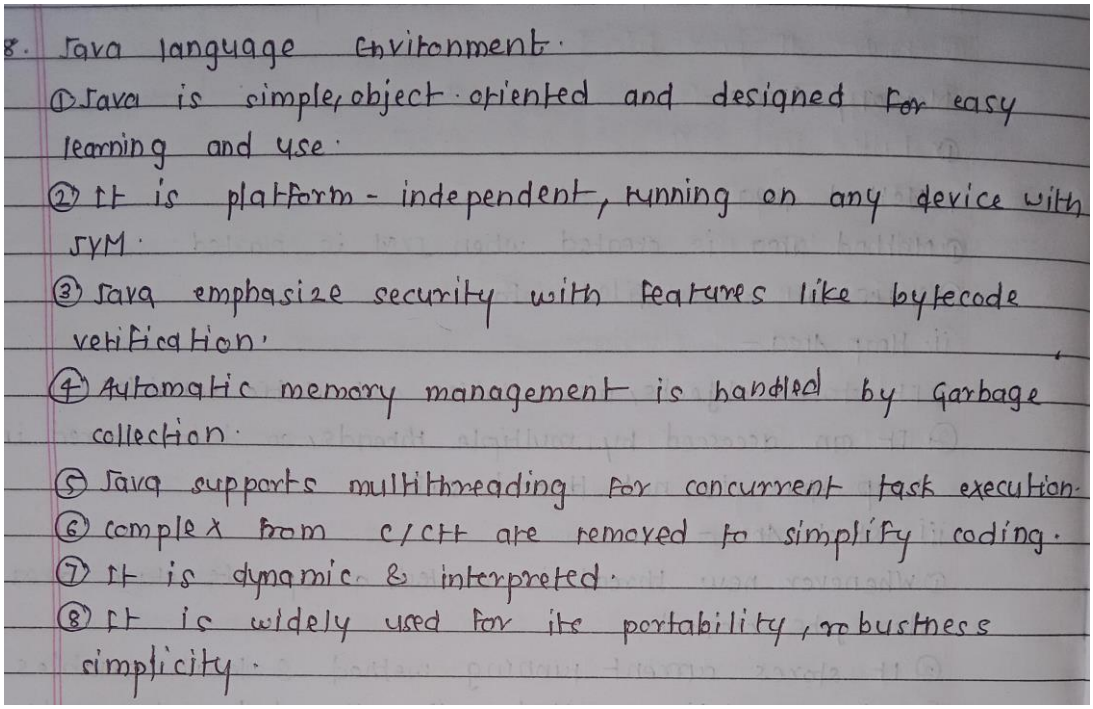
⑥ Class loader is also responsible for Initialization.
i Replace the static value with original value.
ii. Executes the static blocks. First top to Bottom.

2. Memory Areas -

①
i. Method Area -
① Method area is created when JVM is started.
② Stores all class files info.
ii. Heap Area -
① It stores objects, ins. variables and static variables.
② It can accessed by multiple threads, so data stored in heap area is not thread safe.
iii. Stack Area -
① Whenever new thread is created, a seperate stack area will also gets created.
② It stores current running method & local variables
③ After completing method calls, the stack gets empty.
iv. PC Register -
① Next execution instruction are stored.
v. Native Method Area.
All native method calls invole bythe thread will be stored in Native Method stack.

3. Execution Engine -
① Main components of JVM and contains two main parts. i. Interpreter, ii. JIT Compiler.
i. Interpreter interprets the one one line by reading and compile, execute.
ii. JIT compiler - A component of runtime environment that improves performance, by compiling bytecode to native machine code
ⓐ Intermediate code generator
ⓑ Code optimizer
ⓒ Target code generater
ⓓ Native / Mchine Code.
② Garbage collection.

8. Reading Assignment: The Java Language Environment.



8. Java language Environment.

① Java is simple, object-oriented and designed for easy learning and use.

② It is platform-independent, running on any device with JVM.

③ Java emphasize security with features like bytecode verification.

④ Automatic memory management is handled by Garbage collection.

⑤ Java supports multithreading for concurrent task execution.

⑥ complex from C/C++ are removed to simplify coding.

⑦ It is dynamic & interpreted.

⑧ It is widely used for its portability, robustness simplicity.