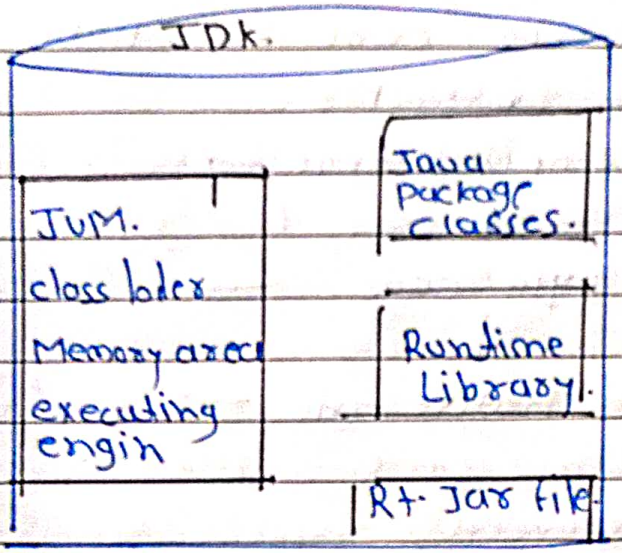
Assignment No 3.

Q1 Explain components of ~~Java~~ JDK.



Java Development kit
it consit JVM Java Package classes.
Runtime library. + RT. Jar file.

JDK also contain Java Development tools.
Java, Javac, Javadoc.

Q2 Difference between JDK, JVM, JRE

→ JDK → JRE + (JVM) + Development tools.
~~JVM~~ ⟹ JRE → it's a combination of JVM class loder, memory area, executing engin. also it has Java package classes. Runtime library & Rt. Jar file which contin executable .class file of our Java program.

JVM → JVM is platform Dependent but The functionality we require ~~T: dr~~ depby our Java code. .class file is same among all JVM on all Platform.

3 Different Role of JVM & How it execute Java code.

→ STeps

Source code → Hello.Java.
↓ compile
Compiled code → .class
↓
byte code → .class.
↓
Native code → readable by CPU code.

So when the .class file goes to the Jvm. ~~Then~~
it enter in Java JVM loder component.
Here are loading of .class file take place.
using three function.
i) Bootstrap ✓
ii) Extention ✓
iii) Application. ✓

where all loading done then .class file goes for linking.
i) varification.
ii) preparation
iii) Resolution.

after that .class files go to method area.
where distribution happens. of
classes method constructer in.
memory section eg like Heap stack etc.

4. →

Execution → There are more but how we use two.

In Execution engin their are two component.

(I) ~~Or~~ interpreter
(II) JIT compiler (Just in time compiler).
(III) garbage collector.
(IV) Profiler

Here code, execution done first using interpreter
Then il compiler find any Pattern like recursion then jit compiler is use for execution.

4. Memory Management system of JVM.

→ as in how JVM works. after class loder. Ther memory area execution start.

Method area. → store class level data such as. class structure, constants, method. characterstic shared among classed & Thread.

Heap → Store object & arrays it's a primary area for bynamic memory allocation.

(I) young generation
— new initilize object

(II) old generation.
- object that survived garbaje collection.

Life time → until when the time they unreachable gc removes that.

stack. → store local variable
method calls, & Partial result.
it created when need after removed

* Programe counter register.
- keeps track of current instruction.
- each Thread has it own Pc register
- life time of thread.
- it have address of next

* Native method stack.
- Purpose = Store state related To native method calls which are method writen in language other than Java (C, C++).

→5 what are Jit compiler & it's role in the Jvm? what is bytecode & why is it important for Java.

→ jit Just in Time compiler in which it can be use To compile Java code which come from ~~Java~~ JVM memory area. but first running method done in interpreter but when ~~int~~ code which repeat itself is find Then it's define go To JIT compiler.

& The execution of these happen repeatedly. these is called. Profiller.

6Q architecture of JVM. → already discused privious.
point (I) class loder
(II) Memory area.
(III) execution engin.

7Q How Java achive platform independency through JVM.

→ JVM is Java virtual machine. present in JRE Runtime enviroment

(I) · class file store in jre → Rt.Jar that class file is responsible for execution.

(II) when any system has JVM it not mean it same for all The platform but The functionality for excuting byte code is same.

So we can say That JVM is platform dependant but it execute Java program using feature thats why Java is Platform independent.

8Q. Significance of class loder & garbage collector in Java.

- class loder → before 3 question.

Garbage collector →
- it identify object which is no longer available. from any live Thread or static reference This is typically done using Technique called reachability analysis. if an object can not be reached by any live reference it's concer eligible for garbage collection
- it's use To deallocate unuse memory

interview Assignement

9.

→ what are the four access modifiers in Java. ~~what is the~~ & how do they differ from each other.

→ Modifier which is used To control visibility of The members of The class / enum / interface.

TOTal 4 Access modifier.

	Same Package.		different Package.		
Access Modifier	Same class	Sub class	Non sub class	Sub class	Non sub cls
Private	A	NA	NA	NA	NA.
(Default) Package level Private	A	A	A	NA	NA.
Protected	A	A	A	A	NA
Public	A	A	A	A	A.

① → A = Accesibal → NA NOT Accessable

Package level Private is / Default.

10 → difference between Public, Protected & detaut access modifier.

→ Public → accessibal from same Package & different Package also (more specific classes).

Protected → ① ~~Package level Private~~

② accessable To same Package & subclass of different Package.

default → ① Package level Private.

② accesbile. In same Package only not outside of The Package.

(11) → can you override a method with different access modifier in a subclass.
eg can a protected method in a super class. 13
To overrideen with a private method in a subclass. 2. Explain.

→ No we can not.
The subclass should be more accessibal Than The superclass.
means it haigher subclass Then we not override.

Accesibility from Top To bottom. / High to low
Public super

Public
Protected. 14
default.
Private. →

12 Protected & default difference.

Protected access modifier accessible only in. same class & same package. & ~~Sets~~ Package. Or different Packg subclass.

15

where
default which is package level Private.
it accessibal only in same Package. →

13 Can is it possibal To make class Private in Java if yes. wher can it be done & what limitation.

→ There are ThreeType of class -
outer clas → Topmost class or superclass
nested class
Method level class.

- Private is define for outer class not allowed.
(*) nested & Method level will be.
- we declare Private.
- only accesibal in That class scope
- if class is nested Then we access in scope class

14 can Top level class in Java declared as protected & Private

→ Private is not allowed - if we declare & how you should acces in another class & not default is not allowed it should be Public always.

it's a Java Thumb rule That all classes should be in Public for super class.

15 what happen when you declare method or variable Private. & Try To acces it in another class with in same package.

→ not Possibal Compile Time error.

16 Explain The concept of Package level Private or default.

→ Package level Private / Default.

(i) it accabile only in same packay

(ii) So we use it only same Package

& it' invoke when no access modifire is deline.

Notes.

Help word.

```
class Abc
{
    Public static void main (String [] args)
    {
        System.out.println ("Hello");
    }
}
```

array symbol.

it's a parameter not a argument

Explain.

(A) class is a keyword.

(B) Identifier

(C) Public is access modifier.

(D) static is keyword.
JVM directly called main method by using static keyword.

(E) Retun type keyword is void.
Void means empty, & nothing not zero & null

(F) main is method which is predefine.

(G) String is ~~cctl~~ class String use because all the Processing done in the form of String. because strings are imutabale.

(H) argument name. (it's array name we can Provide any name at args.

(I) Syntax. class

```
access Modifier class class name
{ // body.
}
```

(J) There can be multipal classe Present. but only one class is Public.

out is static feild declared inside class system

class → it's final class.

(I) system is a predefine class which Present in Java.lang Package

(J) out is Static Variable. Present in Printstrem & in system class Static variable.

(i) IN - out
(ii) out - out
(iii) err (error)

(K) Println is method which is in Present in Printstream.

Printstream contain 3 method

(i) Print() ... Simply Print.
(ii) Println() ... Print output in nextline
(iii) Printf() it use. new line() method.

we can use final, synchronized & strictfp with main method.

file save using Abc.Java.

* variable declared in class. is called. feild
* static access using classname & . operator
* in Java out is called object reference of printstrim. & println strim. it declared in system.

Array & object store in Heap memory.
Stack area store → local variable, current running Process
PC register → store next executing address.
⇒ classes store → method area.

JVM Architecture.

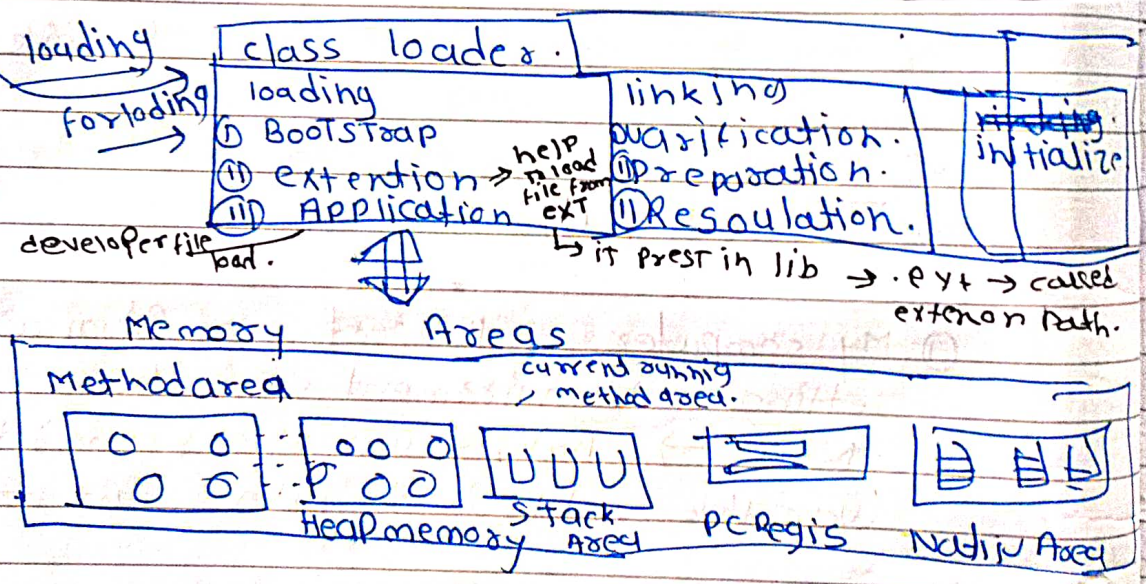
· Java.
↓ compile.
· class
(byte code
↓
JVM

· class file goes To JVM.

* Note
- Heap area store object
- also array stored.
- instance variable.

* interpreter
- read each line one by one & execute it.

* JIT compiler make execution Faster. (method which execute repetadely it convert it into Bytecode using next time it execute



Execution Engin.

NOTE
method which execute again & again we called it Profiler. it present in JIT compiler

Interpreter

JIT compiler
- intermidiate code generator
- code optimizer
- Target code generator.
- native code. (machine code)

Profiler
Garbage collector.

it is a HOTSPOT method

inilization → defalut value replace करती
static फ. original value
↳ it execute all static block. सब निकाल.

Compile & run Java Program by bat file.

direct set name. in direct notepad. using.
classname.bat in notepad.

eg
```
Set path =
Javac Test.Java.
Java Test.
```

bat run
Path set
compie The
run.
Then Pause

using .Bat we hide output But if you want To show Then use Pause command.

- class loder ⇒ (I) . Class file loded To memory Area using class loder → file came from फ. Jar
(II) & loading Process use & Then files load into memory area. memory are stored.

Memory Draw → method area → MA store class level Data, eg fully qualified class name, method, constroctor & all name Store Here, in method area.

Then JVM goes To Heap & crecte class object & store info about loading

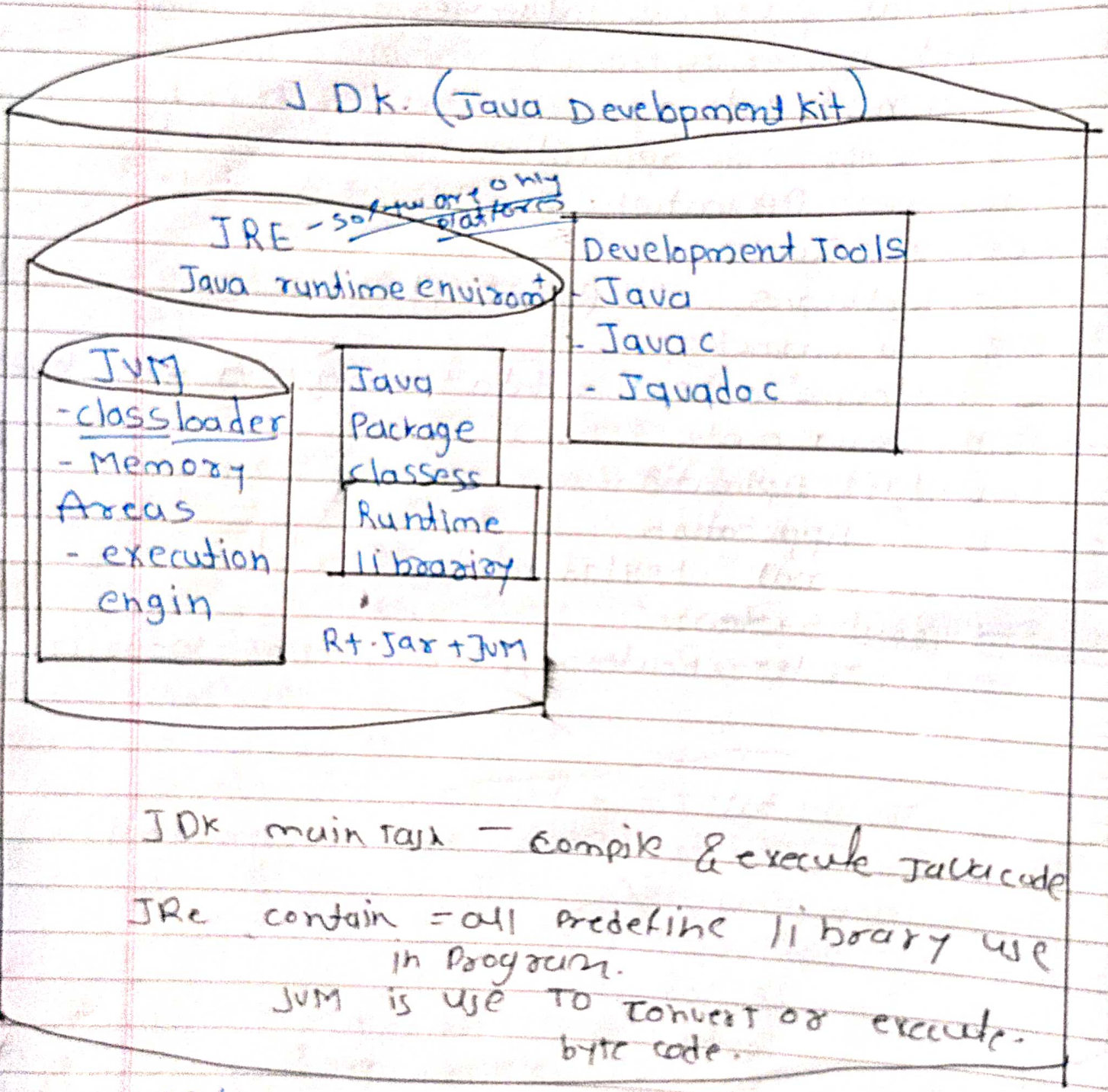
* linking
— varification → it varify How Java files Byte code itis The most used feature. because of These Java issecured

* Preparation — class level data memory में store होता है & initia value assign होती है.

* Resolution → all The refereнce in our Program changes in original refeerences.

compilation = source code → compile → Byte code
JDk
.class
↓
run every where but when The JRE is present.
classmate Date Page

JDk, JRE & JVM Architecture



JDk main task - compile & execute Java code

JRe contain = all predefine library use in Program.

JVM is use to convert or execute byte code.

JDK = JRE + Development Tools

JRE = Java Package classes + Runtime libraaries JVM