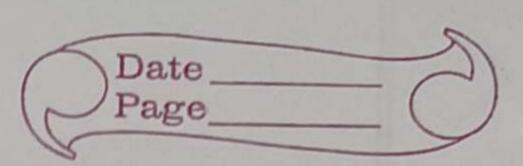
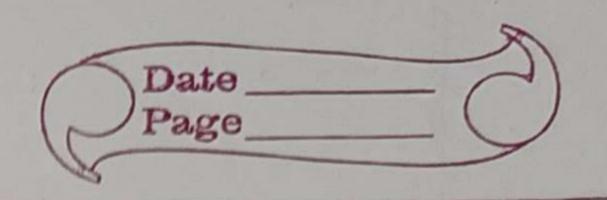
	TNTRODUCTION TO TAVA
	How Java code executes.
	. java file Compiler . class file Interpreter Machine Code (human readable) (entire file) (byte Code) (Line by time 0 + 1 Source Code - Don't runs directly need JVM to
	run this file.
	How Java is platform independent.
	Java is platform independent as the same eal class file (byte code) can own on any device using TVM (Java Virtual Machine) Don't have to one-compile the code for different Operating
•	system "write one, Run Anywhere"
•	But JVM is a platform dependent.
	Architecture JDK = JRE + Development Tools (Java Development kit) +
	JRE = JVM + Library Classes (Java Runtime Environment)
	JVM + Jawa Virtual Machines)
	JIT (Just in time) { Compilez }



	Date
	JOK (Tava Development kit)
•	Provides environment to develop & vuen the Java
	program
	It is a pakage that sincludes:
1,	development itools to provide an environment to
	develop your program
2.	TRE - (Java Runtime Environment) - to execute
	your program.
34	
4>	anchiver - jar
5>	docs generator - javadoc.
61	unterpreter/boader
	JRE (Java Runtime Environment)
9	It is an distallation package that provides environment
	to only vun the program.
	It consist of !-
	1. Deployment technologies
	2. User interface toolkits
	3. Integration dibraries
	4. Bose libaries.
	5, JVM
0	
1	After we get class file, the next things happens at runtime. Class bader doads all classes needed to execute program.
0	TUM conde code to B. t. and ille to execute program.
2.	JVM sends code to Byte code verifier to check format
	of code.



JOK	VS	JRE	VS	JVM	VS	JIT
-----	----	-----	----	-----	----	-----

TDK [Java Development Kit	Total a special descriptions but makes to be used at most of many and an array and a series of the s	
4 provides environment its	develop + vuin	ava sprogram
JRE [Java Runtime Environmen	4]	-> development
1> provides environment to onl		tools
	4 1	-> javac -> Java
JVM [Java Virtual Machine	7	Combiler
	7 -> deployment	-> archiver - jar
JIT [Just-in-time]	technologies	-> docs generator
madagia grattadist	-> User interface	-javadoc
Java interpreter	-tool kit	- Interpreter/
	-> integration	loadez etc.
> Garbage Collector	libraries	ail, and saids
ptc.	> Base libraries	San Andrews
	etc. I de	

the foods of tade should be

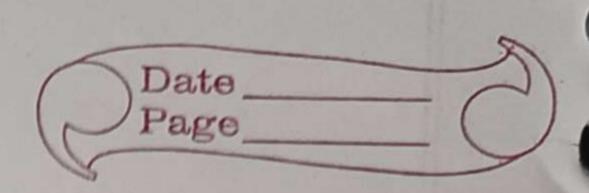
The second the second

-1 restablish anniamil a

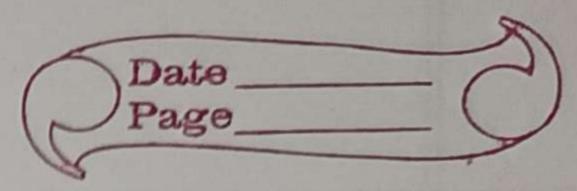
entrole medianistration of the

adidance traville white with TIE

vi- valted ag full- ne act about



	Java Development and Runtime Environment.
	Compile time Runtime
	Compile time Runtime
	(java file) Class loader java
	javac Byte code class (java combiler) vesifier libraries)
	(java compiler) vesifier (libraries)
	class file
	(byte code) (interpreter)
	JYM JYM
	Runtime System
	JVM execution: Java Interpreter:
-	line by line execution Hardware
_	when one method in called
	many times it will interpret
	again & again.
•	JIT:-
->	methods that are repeated,
	JIT provides direct machine
	code so re-interpetation is
	not orequired makes execution faster
0	Garbage Collector!



Moss Inades:-	
Loading :-	
reads byte code file & gener	A SPACE John St. Britain and
an object of this class in	created in heap memory
Linking:	
JVM verifies oclass ifile	
allocates memory for class neplace symbolic neference direct reference	variables & default values
vieblace symbolic vielenence	from the type with
direct vielenence	
Initialization	
all static variables are a defined in the code of state	ssigned with their values
defined in the code of stat	ic block.
Note:-	
Static variable are the vari	inhler donast dohends 11h Ar
	and the dependence of
Object of classes	
JVM gives executable coole	which mune in IRE
Jives executionie cook	
<u></u>	

