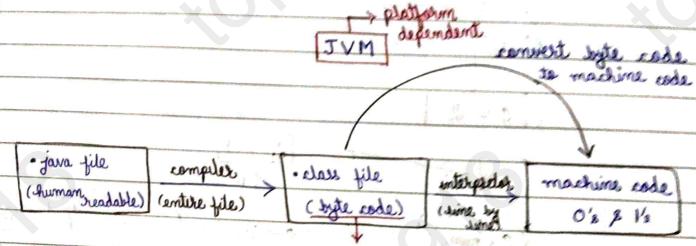
Introduction to JAVA

* How JAVA code executes and more information about platform undependence



- metry gritarego the no nure mas.
- · this code doesn't own directly on system, for this we need JVM.
- * Therefore, JAVA is platform independent *
- ⇒ we can provide this byte code to any system means we can compile the JAVA code on any system.
- ⇒ But JVM is platform dependent means for every 0.3,

 the oxide executable file we get, it has step by step

 set of instruction dependent on platform

DK (JAVA Development Kit) DK (JAVA Development Kit) DR (JAVA Rundume Envisionment) DR (JAVA Abendament the burn thin to pagram and the labele mint Jak (JAVA Abendament) Jak (JAVA Abendament) Jak (JAVA Abendament) Java			
K (JAVA Development Kit)	A long to	JRE " JVM	
(5 (5 AVA Rundume Emissionent) 5 DK (5 AVA Development Kit) 7 T T 5 Development Kit) 7 Letus 5 Development Kit) 7 Letus 7 Development 7 Development 7 Development 7 Development 7 Development 8 Development 8 Development 8 Development 9 Development 9 Development 1 Development	岩	(JAVA Development Kit)	60
JDK (JRVA Davelopment Hit) + deployment Jackmelogies Jackmelog	2	(JAVA Rundlime Emvironent)	1
TTTT + Journal Spetal + web interlebrace + Journal + Jou	A B	(JAVA Development Kit).	Tear
TITE in the family contacted to the intersperate to the intersperate to the intersperate to the family contacted to the family			4
The fundament of the first of t		gradespoten -> wer interferen	1
		- Gallector integration.	
	<u> </u>	23	Chiamac
			1
	200 500 4g		

JAVA Development and Runtime Environment suntime. compile time class loader · java dava file byte code verifiet class javac. Merariel (java compiler) AVAT JIT JVM apple . Interpreta Cobyte code Runtime System ⇒ JVM Execution Hardware · JAVA Interpretor * class loader :-→ line by line execution -> when any one method · loading is called many times it will interpret again 8 alig aboa tyte code file 8 & again generate binary data -> an object of this class is stored in heap. · JIT -> method that are repeated, JIT provides · dinking → JVM veryles · class file direct machine code

· Jarbage Collector

seguired.

la re-interpretar is not

-> makes execution faster

· Initialization

-> all static variables are assigned with their value

-> allocate memory for class

-> replace symbolic reference

variable for and default value

from the type with direct sef.