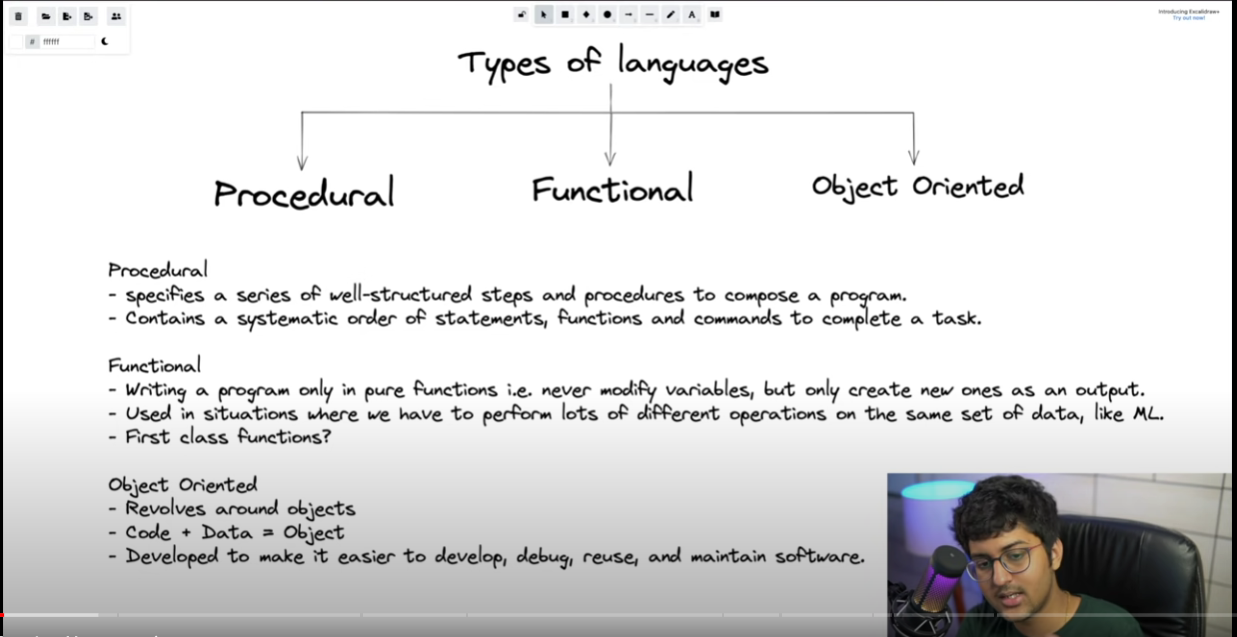
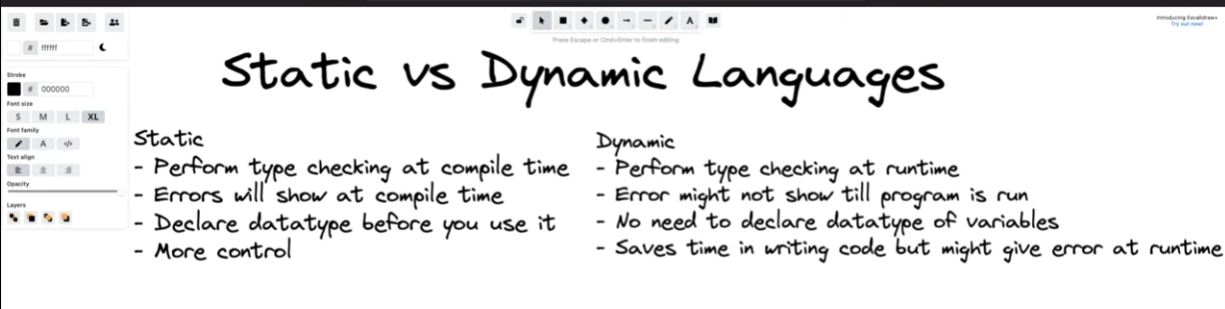
**Community Classroom**

Lec1 :





Let say

A = 10

A= “soumya”

It will not create an error for dynamic language

Object reference variable:

There are two types of memory

1. Stack memory
2. Heap Memory

A=10

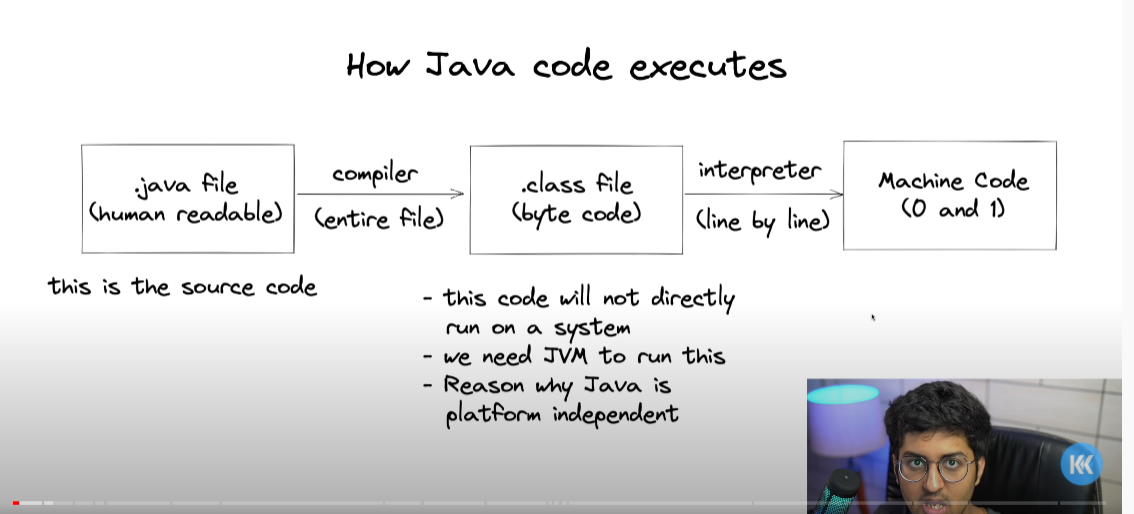
A = reference varstored in stack memory and 10 object stored in Heap

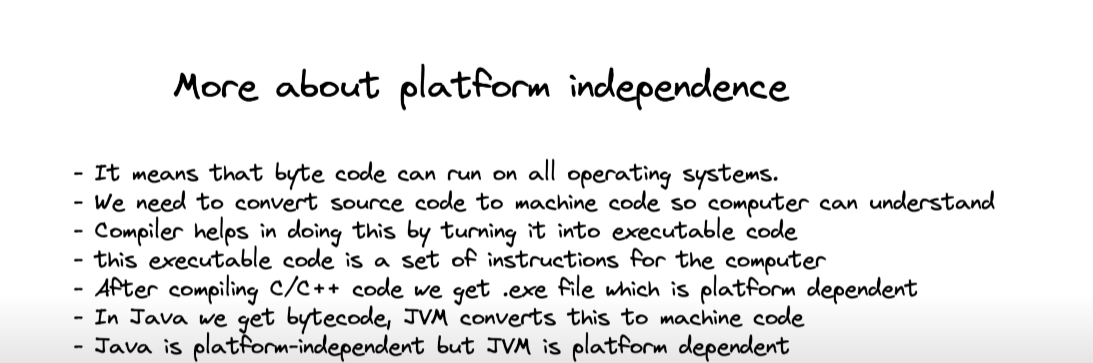
Garbage collection : Memory space which is not pointed by any other variable

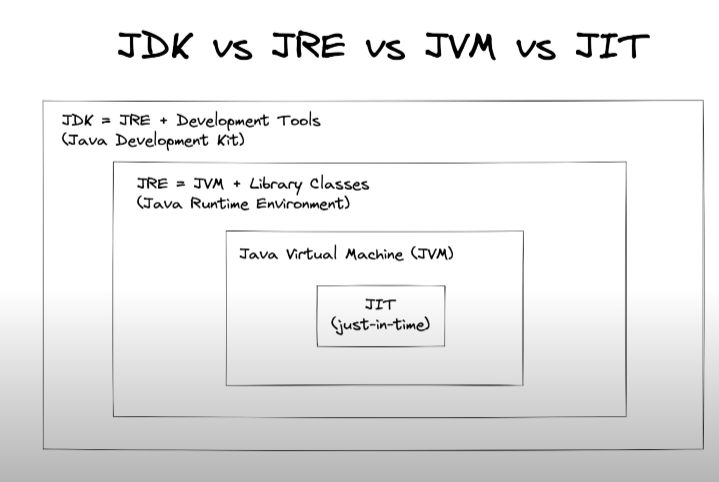
**LEC2 : GITHUB and Use :**

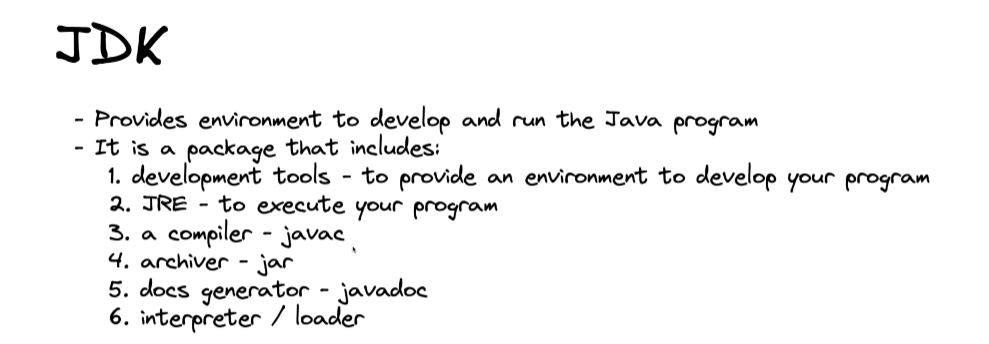
|  |  |
| --- | --- |
| **1** | To check Git is installed or not :  Command : git  o/p : |
| **2.** | Unix commad :  ls ,  ls –a (for hidden files /folders) |
| **3** | Git init : to start git repo using initialisation  git config --global user.email [soumyadeepsen97@gmail.com](mailto:soumyadeepsen97@gmail.com) : to initialize gmail and account setup  Git status : to see all the newly added commands  Git add . : to add all the files in current directory  Git add abc.txt : to add abc.txt files into github  Git commit –m “the comment” - > to commit the changes  Git log : for all the changes  Git reset <logid > : torestore the files till exact logid  Git stash : to store all the previously deleted files in uncommitted to stored file  Git stash pop : all the previously commited things are again configured  Git stash clean : clean  Now connect with git repo :  git remote add origin https://github.com/soumyadeep28/DSA-HUB.git  : this command will connect the repository (origin is the name of url in short)  git push -u origin main : to push the files to git  git remote –v means : what all the links are connected |
| **4** | Git branch <newbranch\_name> : this is to create new branch    Git checkout <newbranch\_name> :it will switch to new branch from master  HEAD points to the new branch  Git merge <<newbranch\_name> : to merge with current node  To switch -> add ->commit ->checkout to next node -> merge |
| **5** | To create a copy of a git file to a new file in own git store :Use Fork (in the browser)  Git clone <link> : to download the files in local files |
| **6** | To merge the own project branch with mail project branch : pull request is used |
| **7** | One pull request associate with one branch (if a branch initiated a pull request it will not allow 2nd time the pull request)  Git fetch - - all - -prune -> to fetch all the changes in that branch  Git reset - -hard upstream/main 🡪 this will update the all the update in the (main branch) |
| **8** | Git rebase –i <log\_id> : it will give the all the comments used for commit  this will merge 1 comment with 2 , 3 , 4 etc . this operation is swash (merger) and pick will use to segregate the comment  To exit from there  Esc + : + x + enter |
|  |  |

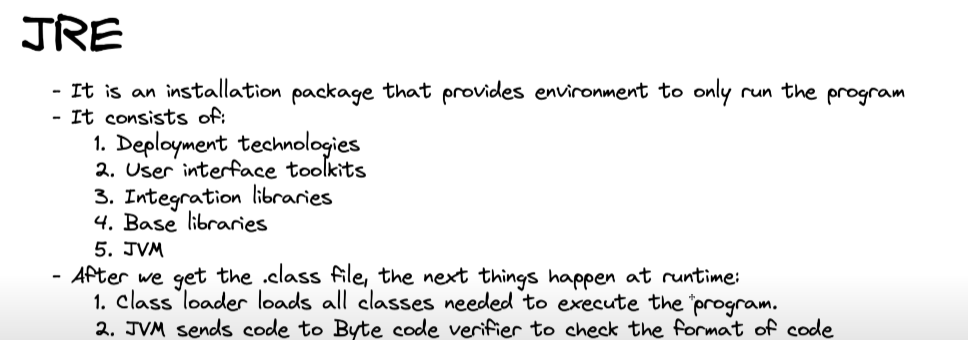
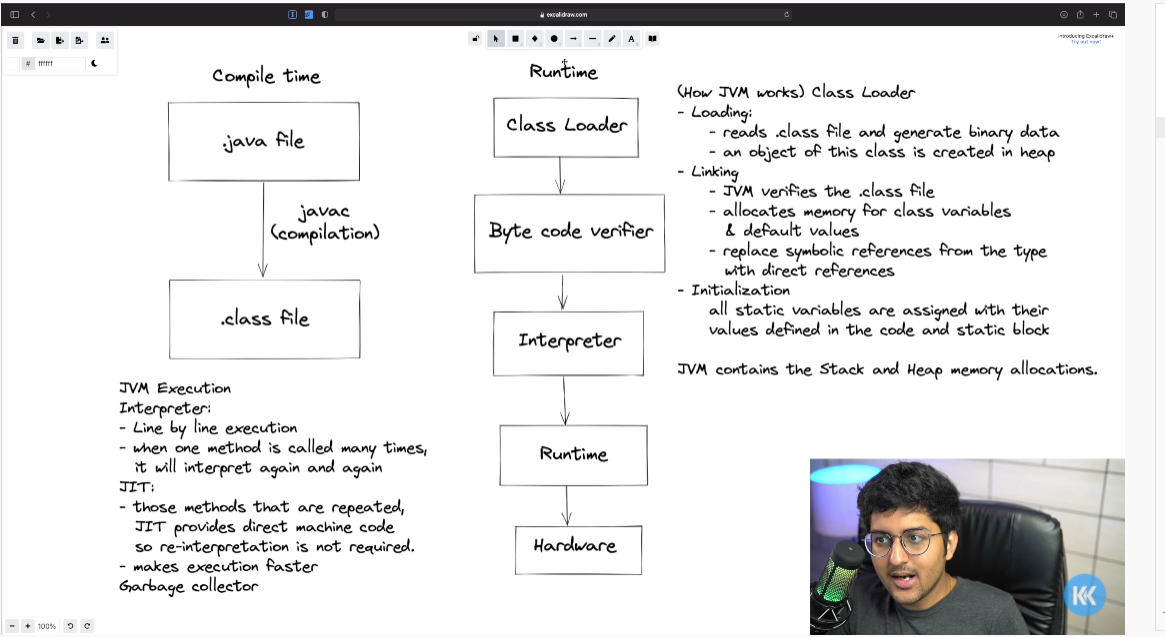
**LEC3 : Introduction to Java :**

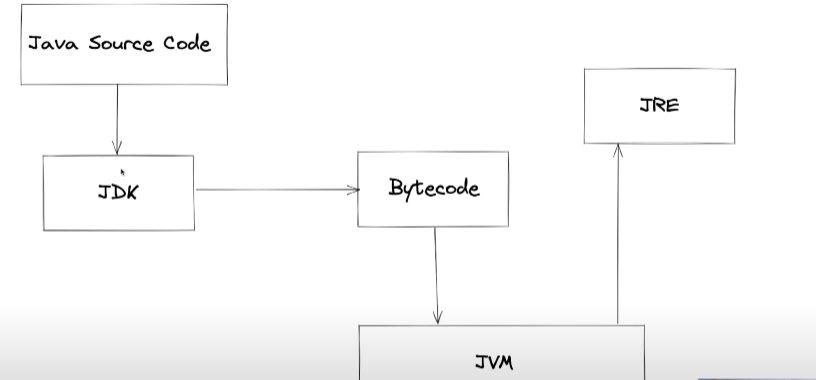










# LEC4 : First Java Program - Input/Output, Debugging and Datatypes:

# We are creating a Main.java

# Note : each java file (i.e abc.java means abc is a class mentioned in the file)

# Class first letter name of the class should be capital

# Access modifier a. public b.