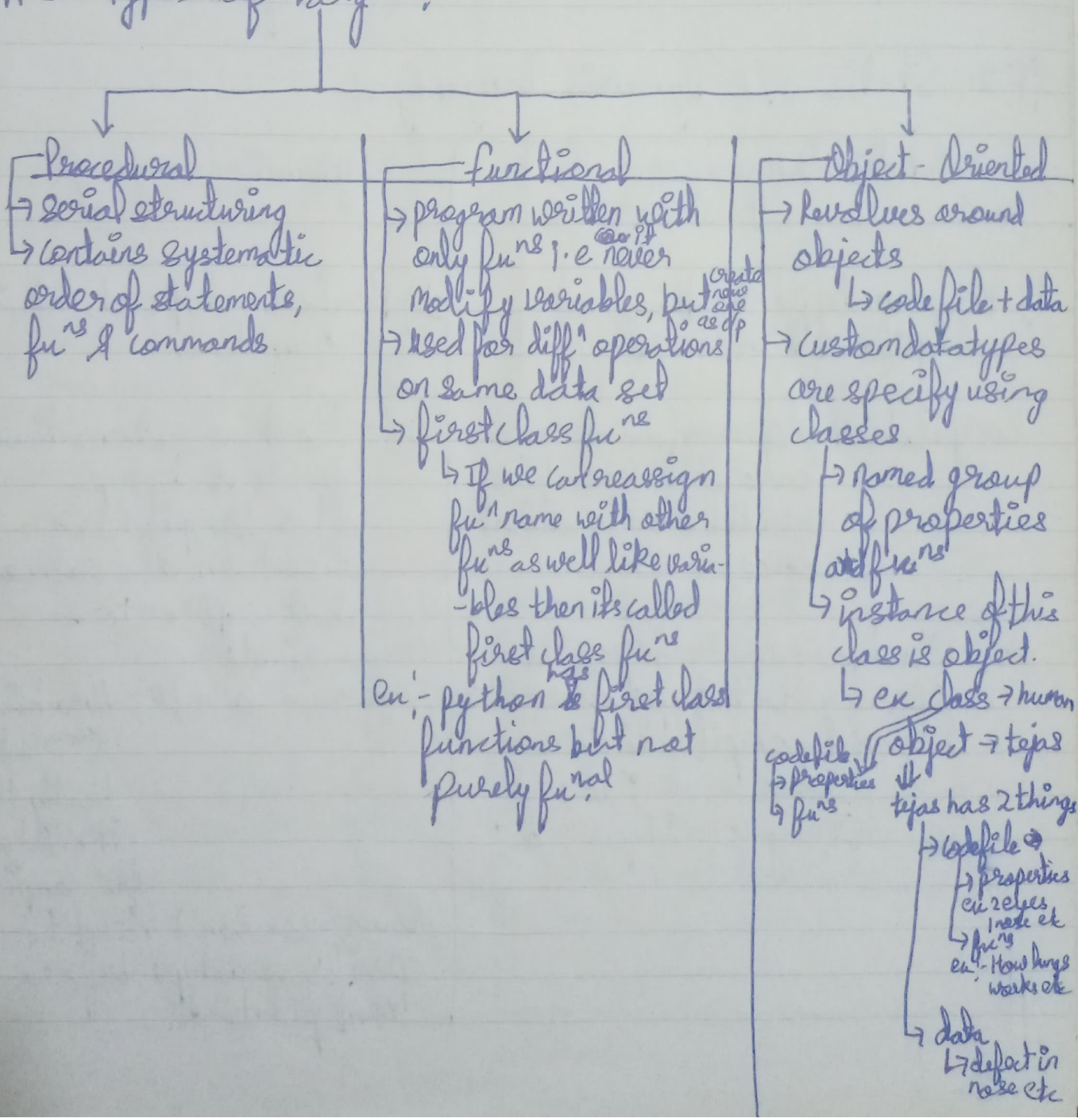
Java + DSA Bootcamp — September 8, 2021

★ Lecture 1: Intro to programming - Types of langs, Memory Management

#1. Types of langs :-



↳ Divided code in diff" chunks to make it easier to develop, debug, reuse & maintain software

#2. Static v/s Dynamic languages :-

Static languages	Dynamic languages
1) Perform type checking at compile-time.	2) Perform type checking at run-time
compilation ⇒ converting of source code into machine readable code all at once.	eg:- Machine automatically decide datatype after compilation based on the value that is given to a variable.
eg:- Machine required you to specify datatype of variables before compilation	eg:- a = 10; // Machine know automatically that it is integer after compilation
eg:- int a = 10, b; String c = "Tejas", d;	Means, you don't need to specify datatype before compilation.

2.) Errors will show at compile-time.
ex:- int a = "Tejas";
// code doesn't compile and gives error that you can't store String to integer.

3.) Declare datatype before you use it.

4.) In static languages you can't store, update and string to integer variable and combination like this
ex:- int a = 10;
a = "Kunal";
// Error at compile-time
b = 'a' + 10; // also gives error

5.) More control
// Runtime errors are reduced.
// More time in writing code.

2.) Error might not show till program is run.

3.) No need to declare datatype of variables

4.) In dynamic you can store and update any variable to other datatypes.
ex:- a = 10;
a = "Kunal";
because of dynamic property & reference variables (discussed later)

5.) Saves time while writing code but may might give error at runtime.

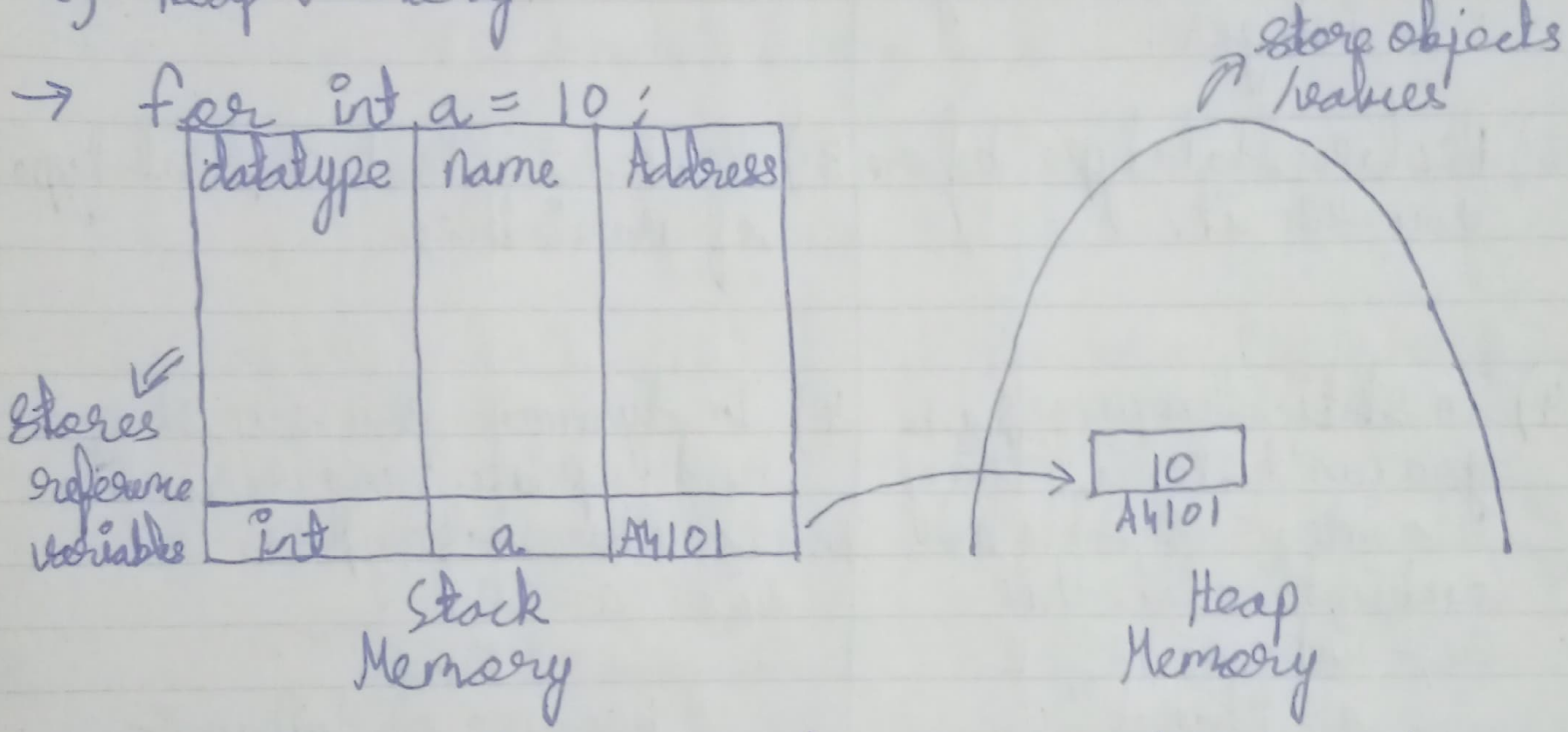
#3.) Memory Management :-

→ 2 portions in Memory Management

- There are 2 types of Memory ;-

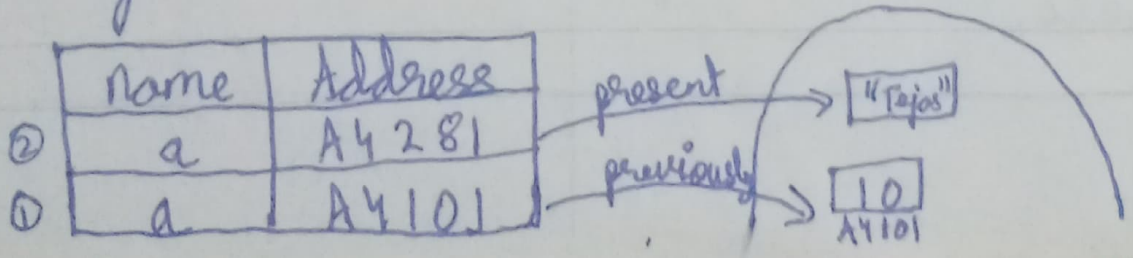
1.) Stack Memory
2.) Heap Memory

→ for int a = 10 ;



Note ;- In case of dynamic languages datatype column is not there in stack memory that's why we can assign any type of values/objects in any variable and also update them with diff^n datatype values unlike static languages

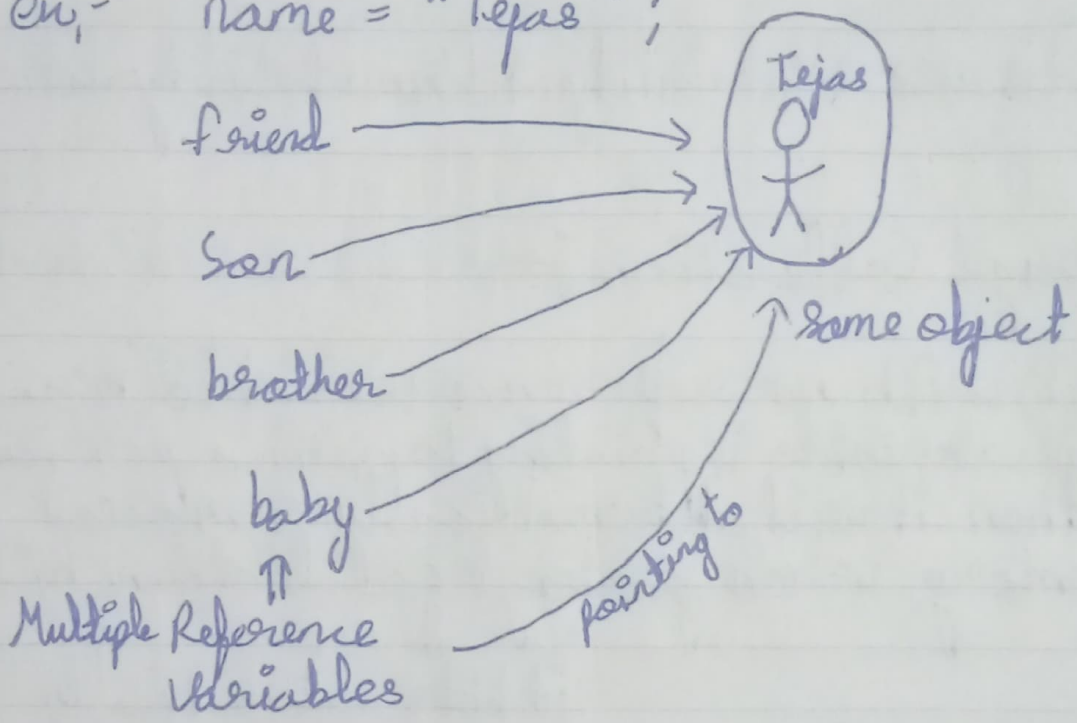
ex ;- In static, below case is not allowed but in dynamic it is allowed.



#4.) Some Important properties of Memory :-

1.) More than one reference variables can point to the same object!

eg:- name = "Tejas";



2.) If any one of these reference variables change the object, then original object is going to be changed and going to be changed for all.

eg:- Suppose in above example, If Tejas change clothes then all can see the new clothes, even when change done by 2nd reference variable i.e Son
Mother said Son i.e

This topic is Important cos java only has pass by reference value during fn" parameters and we do both pass by value & pass by reference using pass by reference value.

Note;- We discuss Mutability & Unmutability later on.

#5.) Garbage Collection;-

If an object with no reference variable (means no reference variable is pointing toward a particular object), then it will automatically removed from memory when garbage collection hits.

⇓

it hits automatically

Note;- We can see later, that How to perform some functions when garbage collection hits in Object-Oriented programming.

↳ There is finalize etc concepts...