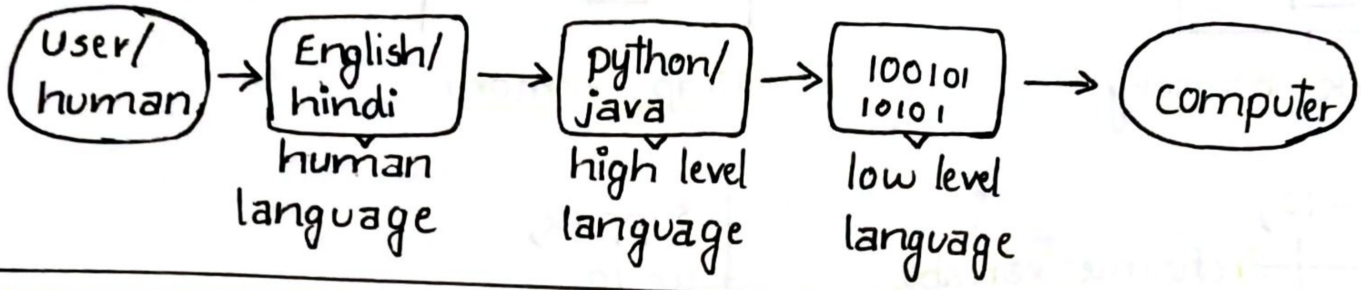
Introduction to programming

Computers natively or at the base level only understand or accept bits (0's & 1's).



Types of programming languages:

Procedural :- properly structured procedures or steps to form a program.

- step-by-step order of commands or code.

Functional :- pure functions are written. variables are never modified, new ones are created as an output.

- used when we need to perform different operations in large quantity using same set of data. eg: ML

Object-oriented :- Object based, code + data = object

- was developed to make it easier to develop, debug, maintain & reuse.

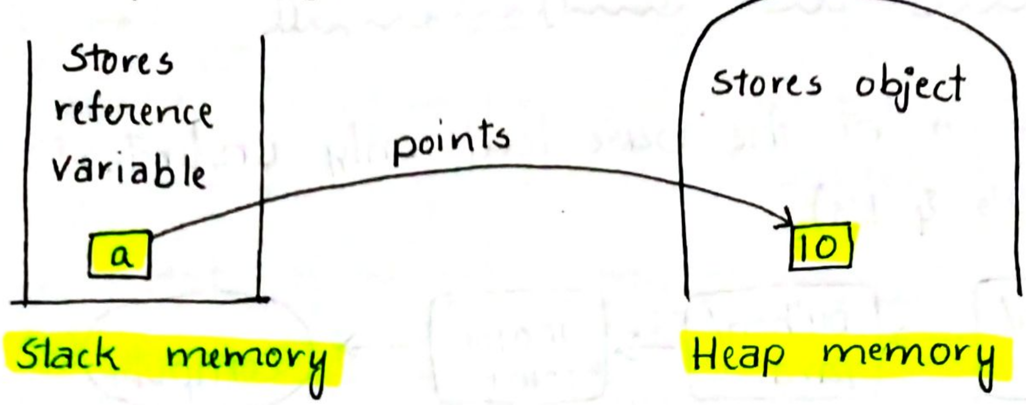
Static Languages

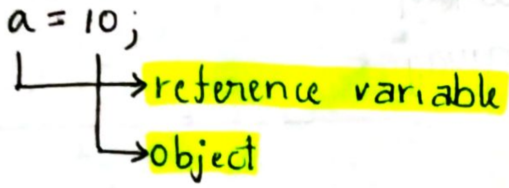
- perform's type checking at compile time.
- error shown at compile time
- need to declare datatype manually. eg: int a = 3;
- more control over programming but time consuming.

Dynamic languages.

- perform's type checking at runtime.
- error might be seen till runtime.
- datatype is declared automatically. eg: a = 3;
- time saving but chances of error at runtime is high.

Memory management:





Suppose,

a = 10

b = a

then $a \xrightarrow{2} 10$ = $b \longrightarrow 10$
$\overset{\uparrow 1}{b}$

∴ b = 10;

- more than one reference variable can point towards one object.
- If the initial reference variable's object is changed then all the reference variabes pointing to the same object ~~will~~ value will change. eg:- a = 10

  b = a ∴ b = 10

  Now, a = 50

  ∴ b = 50

- 10 is removed with garbage collector since it does not have a reference variable.