PYTHON

**Variables**

Variables - They are like placeholders to store something

Variable name – to name the grouped set of something

Eg:

A= 10

1. Variable name

10- variable

**Datatypes**

Integer type

String type

**Operators**

Addition

Division

multiplication

**Casting**

Converting one datatype into another datatype

Eg

A=”10”

B=”20”

C=a+b

Print(c)

o/p: 1020

(because it is in string)

Here casting can be worked as

A=int(”10”)

B=int(”20”)

C=a+b

Print(c)

o/p:30

**How to get input from user**

A= input()

If it is integer

1. Int(input())

**If else with Boolean values**

**Boolean values – True ,false**

If (true)

Print(adsdf)

Else

Print(sdnfsd)

**Comparison operator**

Print(“win”==”winn”)

% 🡪 modulus

/ 🡪 divide

% 🡪 provides remainder

/ 🡪 quotient

10/2 🡪 5

10%2🡪0

**Nested if**

**For loop**

**Nested for**

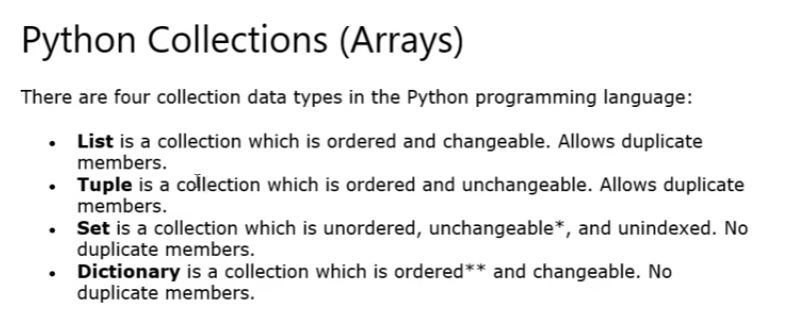
**While loop**

**Python collections**

List

Tuple

Set

dictionary

functions

return keyword

classes and objects

if no class and obj:

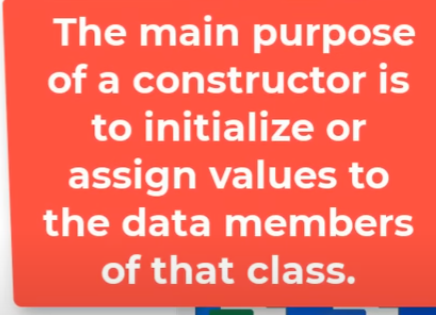
1.requires large amount of variables

2.code length

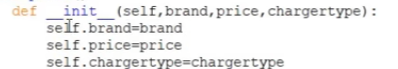
3.code variable confusion

It will help for code efficiency,code reusability etcc

Constructor and self keyword

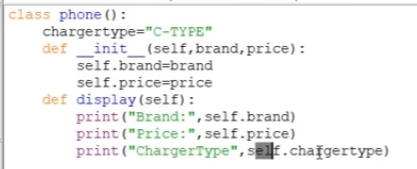
  
class variables

Instance variable



Self. Everything --🡪 instance variables

Class variable



Below class there is a declaration where it specifies the same data to all the functions

Inheritance

5 types

Super class

super().\_\_init\_\_()

#super is used to call the constructor from parent class

Polymorphism

Method overriding

Encapsulation

\_\_company name 🡪 private variable

\_ 🡪 protected

\_\_ 🡪 private

.companyname 🡪 public

Exception handling

Compile time error – during compilation it may be name error, syntax error etc

Logical error – a+a some kind of logics

Runtime error – input la 1 no 2 string 🡪 value error