Inheuit mce In heritance Single level Multilevel inheriten a Mulhiple level Coronel fallo Povent pavent fater Lild child. Inheusterace 1) class User: def login (self): print ("Login") det sugistration (self): point ("Registration") Class Student (User): def enroll (self): found ("Enroll") det review (self): print (Review) crease object

Sty = Student () sturemad() Of Emall sty registration ()

OIP pe gi stradian

def --init- (self, price, brownel, comera)?

def --init- (self, price, brownel, constructor")

self. price = price

self. brownel = brownel

self. comera = cormera

dof buy (self)?

print ("Buying a phone")

def velum- phone (self)?

print ("Returning yshone")

Class Smootphone (phone).

```
8 = 3 marfphene (20000, Vivor, 15)
     Inside phone Constructor
 # Inheastiting private members
The healting constructor.
  class Phone:
      def -- init- (self, price, brond, camera):
            print (" Inside phone constructory)
            Self. -- price = price
            self brond - brond
             Self. corners - Comera
      dof buy (self):
           provid ("Buying a phone")
      def suelum - phone (self):
           porint ("Returning phone")
  Class Smartphone (phone);
  V = Smartphone (45000, "OPPO", 20px y
 Insi de phone constructor
    Ve Phone price
         019 45000
It Inheritence Constructor
class phone:
         def--init-- (self, price, bornol, cornero).
              print (4 Inside phone combuctor")
                self. -- price = parice
                 self. brand = brand
                solf. Campora = comosea
```

def buy (seg) point (Buying a phone ") de f noturn-phone (self); point ("peturing phone") Class Sonautphone (phone) def buy (seff): create object (Buying asmartphone is) V= Smaelphone (4000, 40000, 40000) OP Ins, de phone Constructes v. buy () Of Bying a smeatphone # single level inheriterace class Parent: def -- init - (selfgmum): self. -- num = num defiget-num (self): return self. -- num class Child (Parent): def show (self). create object print (uchild class ") point (son. -- num) (It shows error because num attribute is private) Son= child (55) sun show () Child Class

It Super Keyword daughter = chile Nef -- init - (self, price, brand, camera); class Phone: daughter ge print ("Inside phone constructory) 0/2 400 daughter. self. -- price = price Essas - Co self. board = board class phone Self. Comera = Comera deg bery (self): point ("Buying a phone") def neturn-phone (self) point (" Returning phone") def Class Smartphene (Phone); def by (self); class mut ("Buying a smaetphane")

super(). buy()

v = Smartphone (55000, "V,vo", 30px") Of Inside phone Constructor) call buy() Il Buying a smartphone Object Buying aphone. 1) class Parent: def -- init -- (self 9 num): Inside most self-wine - wine 0/15 des get-num (self): rogn class child (Pavent): OPP def -init - (self, val, num); self -- val = val def get-val (sel): return self. -- val

off Super Reywood Chughter class Phone: ere f -- init - (self, price, borne, carnery); daught print ("Inside phone constructory) 0/2 +00 daught self . _ price = price Error. Self brand : bound class Selfo Comerce - Comerce def bey (self): print ("Buying a phone") def notion-phone (self) point ("Roturning phone") Class Smartphone (Phone); deg by (self); must ("Buying a smallphone") create Object Super(). buy () V = Smortphone (55000, "Vivo", 30px") Of Inside phone Constructor) carl buy() Ilaying a smertphone Object Beying aprine 1) class Parent: def -- init -- (self mum): Selfemum - num des get-num (seg): class child (Pavend): def - init - - (self, val, num). self -- val = val def get val (sel).

Super Keywood class Phone: Nef -- init -- (self, price, board, camera): print ("Inside phone constructory) self. -- price = price Self. boarnel : boarnel Self. Comera = Comera def bey (self): point ("Beijing a shore") def neturn-phone (self) point ("Returning phene") Class Smartpheno (phone); def by (self); point ("Buying a smaetshone") create Object super (). buy () V = Smartphone (55000, "V, vo", 30px") Of Inside phone Constructor) cell buy() Il Duying a smartphone Beying aphone 1) class Parent: def -- init -- (self , num). Self-worm - som dej get-num (self): selven self num class child (Parend): def -- init - - (self, val, num). self -- val = val def get-val (sel): refuon self. -- val

Laughter = child (100,50) daughter get-val () daughtfær. gel-num() Error - child object has no attoibute num! class phone? def -- init--(self, posice, bromel, cornera): point (M Inside phone Conston der 11) self. porice = porice self. Isomel = bosomd zelf. comera = cornera def buy (self): point ("buying a phone") class Smootphone (Phone); def -- init -- (self posice, board, comera, 05, ram): super () · -- init - - (poice, board, cornera) self. 05= 05 self. seen = sour point (" Inside smartphone constructor ") S= Smootphone (20000, "Apple", 15, "Android", 2) Of Inside phone constauctor Inside Smootphone constructor. port / s. comera) roin (S. boomel) Of Ppple

Multilevel Inheuitence class Product: def review (self); point ("Product automor review") class Phone (Product): def _ init - - (self sporce , borond, comera). point ("Inside phone constructor") self. -- ponce = ponce self. brond - brand self. comera - comera def buy (self): point (Buying a phone 4) Class Smoot phone (phone): 3 = Smartphone (20000, uppsley, 15) Il Inside phone constructor S. review () of product customer roview greate Object (1000000, VIVO 4,12) OIP Inside phone constructor p. review() Off product customer novices.

Inheritence_ H-Mulliple class phone: def --init -- (self, porce, lorond, comera). point ("Inside phone constructor") self. -- price = porice Self brand = brand self. Comera = comera def buy (self). point (Buying a phone) class Product: def review (self): print (Product customer noview") class smartphone (1000000 "VIVO 1/2) Object pass = Smartphone (1000000, "VIVU", 12) They de phone constructor p- review () OIP Tooduct customer neview Static Variable Class Afon: def -- init -- (self); 3elf. --pin = 411 self. -- balance = 0 self. sno =0 self-sno += 1 def mener (self): Sbi-Afm() 861 = 1 [m() unb (NIMI)

Example of Aggregantion (Symbol - 1) Class Customer: aleg --inet (self, name, gender, address. self nome - nome self. gender = gender self- address = address Class Address: deg -- init - (self, city, pinno, state); self city - city self. pinno = pinno Object

a = Address ("Kolkalan, 700006, uwest Berge) (= Censterner (upriya 1, utemale 4, 9) Offine (1. address

-- main -- . Address object at 0x0 econ of Pojymorphism The word polyonorphism means having many ! forms. In progressming, polymorphism means the seme function more (but different signatures) being used for different Jugses. 9) Nethod Overviding is) rethod overloading in steemed overloaded

```
Mothod Oresouding
  class A:
    def ml (self)
          return 20
  class BCA):
       def micself);
       dej m2 (self):
           seturn 40
   class (B):
      ref m2 (self);
    05g1 = A()
    06/2 = B()
    0好3二(()
   point (06jt.mx1)+06j3.mx1)+06j3.m2())
OP 70
  method overloading
   class Shape:
       def area (self, v):
           refurn 3.140 och
       def area (self, a, b):
           return a xb
      p = Shape ()
        Error (Python doesn't support overloading)
       p. area (2)
mother approach
    class Shope:
         det avea (self , a, 8=6):
                 return 3.14 xaxa
                  referon axx
      s = Shapa
         print (s. area/2)
```