abstract class Parent {

int a;
int b

Parent(int a, int b){
this.a = a
this.b = b
}

Void display();

void authentication(){
if(this.a == "credential"){
}
}

→ new Parent()

class Child extends Parent

int a;
int b
int c;
int d;

Child(int a, int b, int c, int d){
super(a,b)
this.c = c
this.d = d
}

void display(){
S.O.Pfl(this.a, this.b, this.c, this.d)
}
}

a b c d

new child(1,2,3,4)

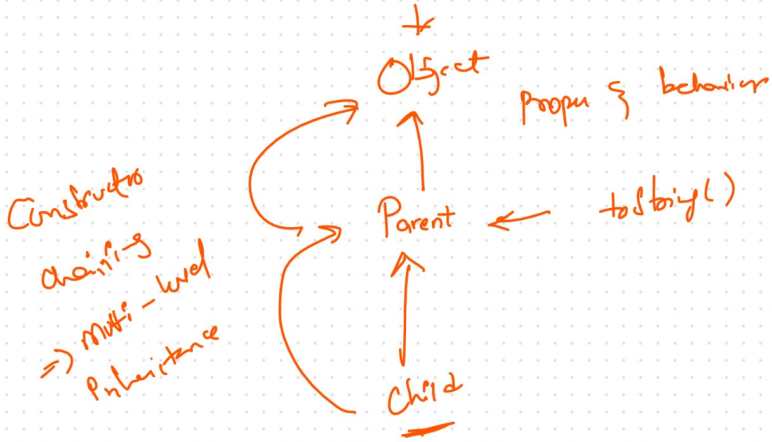
Constructor Chaining (Cont):

→ It helps in inheriting all the properties & behaviour from the object class [multi-level inheritance]

Object { }
↑ p & m

abstract class Parent {
Parent(){
super()
}
}

class child extends Parent {
child(){
}
super()
}



Q: [Home work]

① Need of constructor in abstract class // to initialize the instance variable & which is return can inherited from parent class

② Role of Constructor chaining