

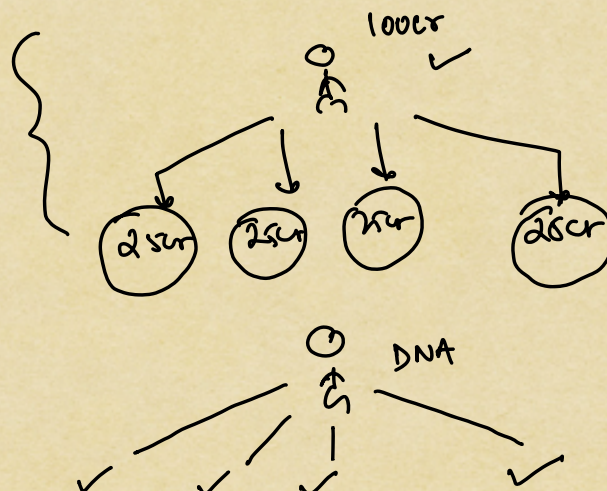
## ⇒ RECAP

- constructors
- access modifiers
- deep copy shallow copy
- pass by value → address

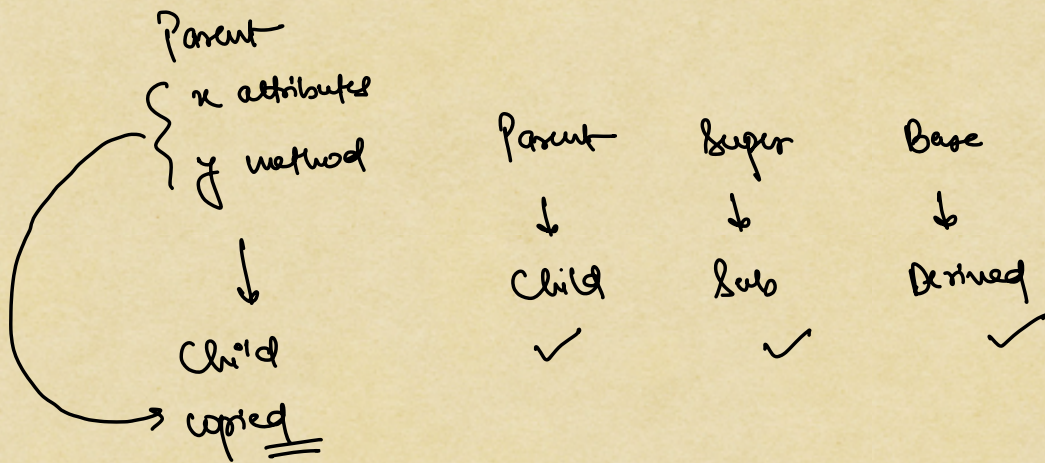
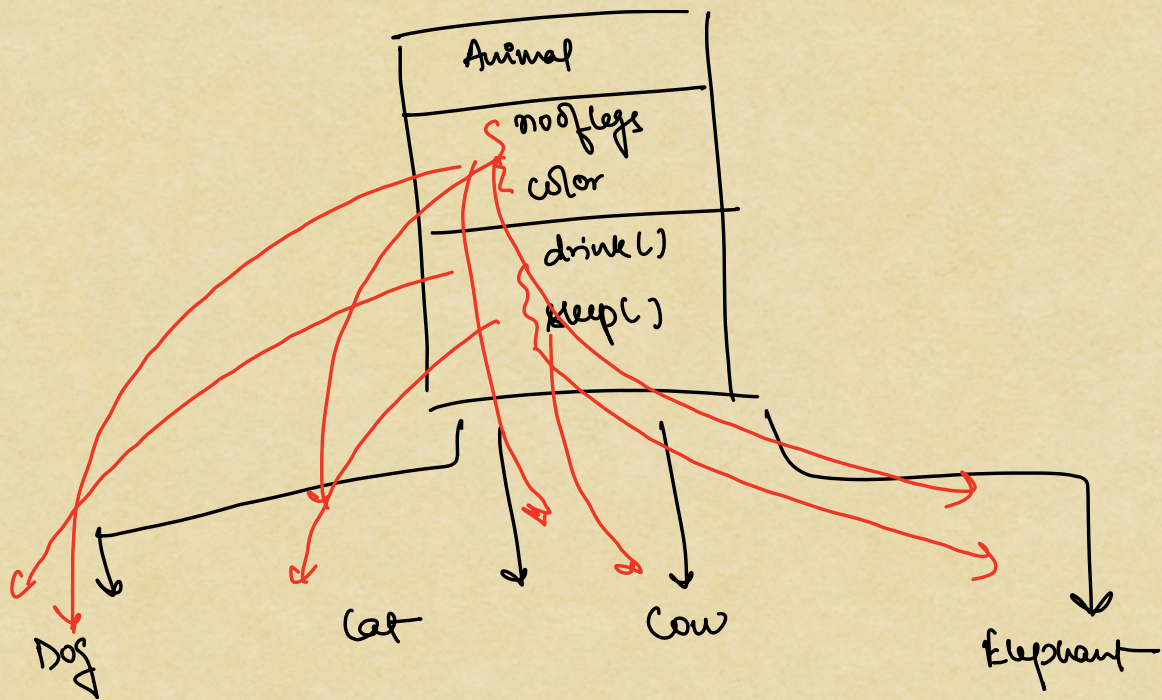
## → Agenda:-

- Inheritance
- Constructor chaining
- Polymorphism
  - overloading
  - overriding

## ⇒ INHERITANCE

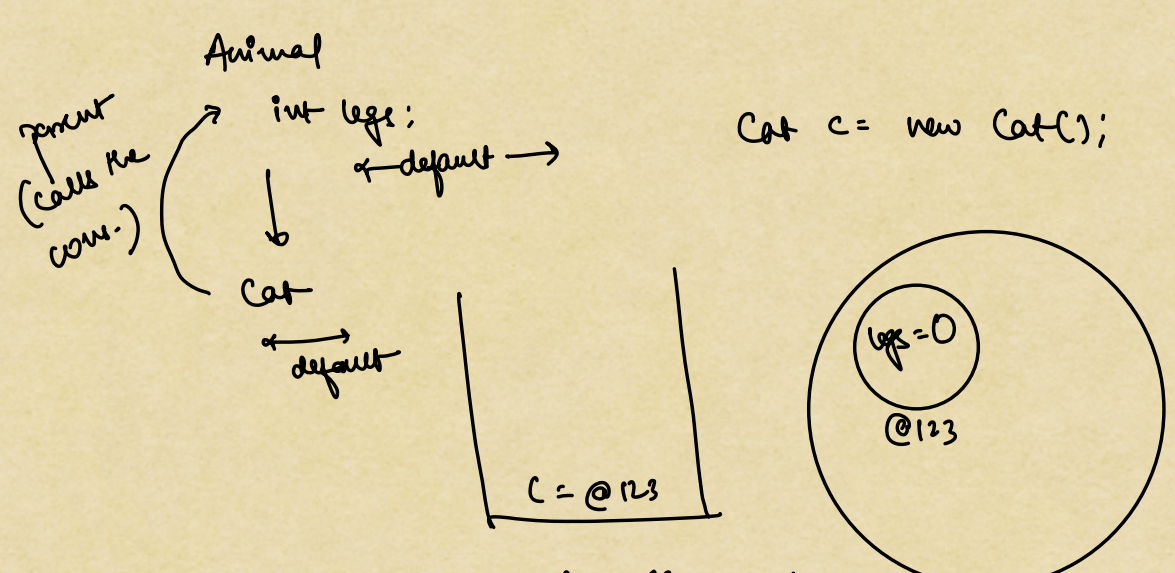
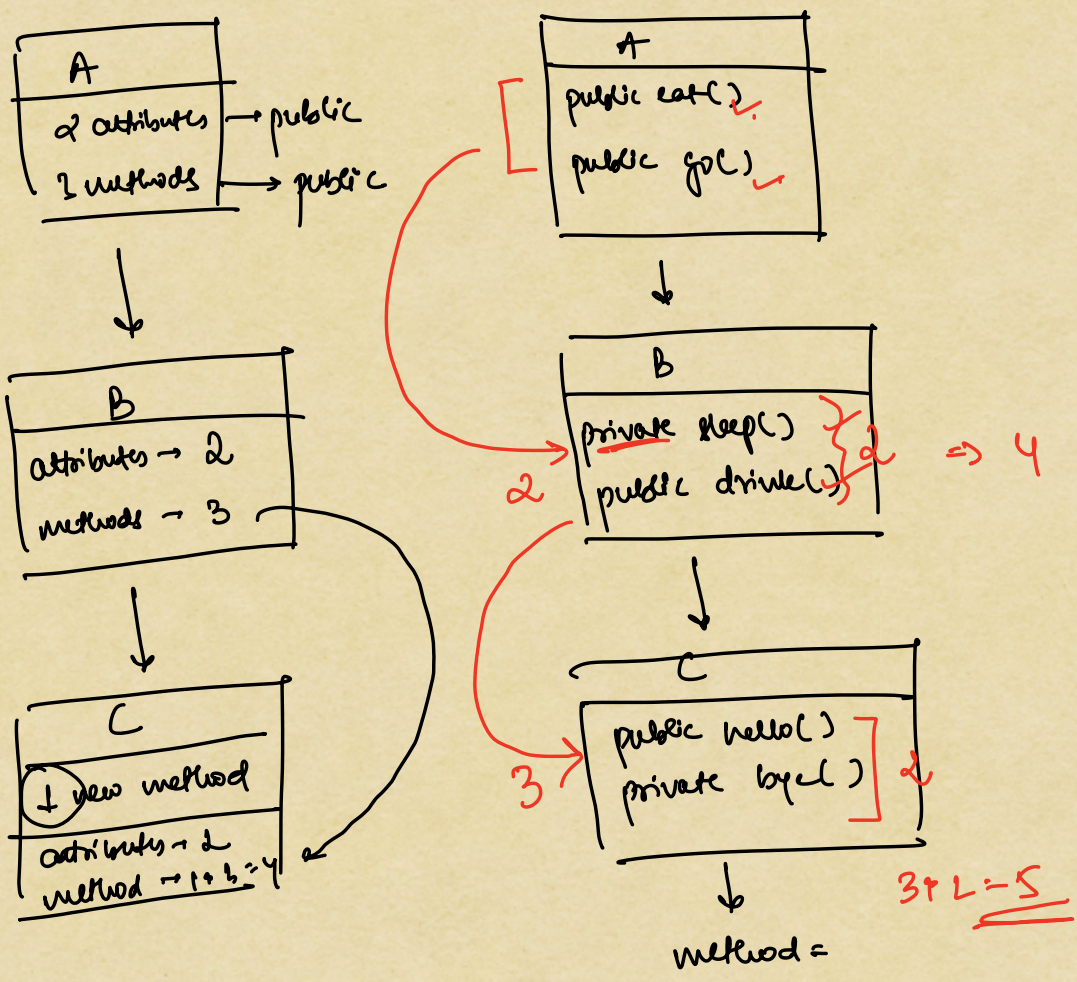






\* private members have no advantage via inheritance

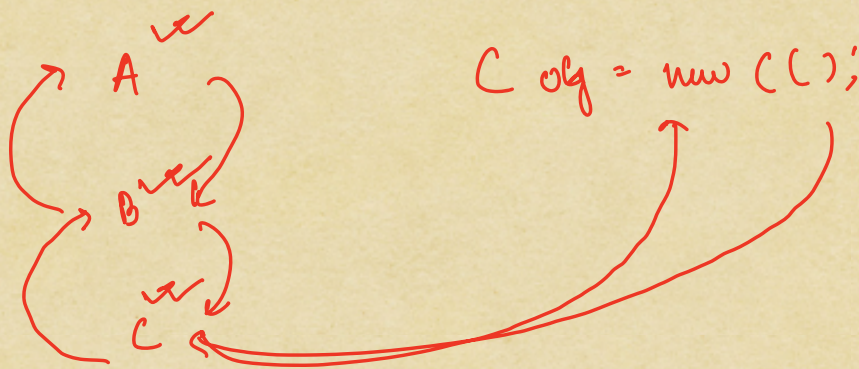
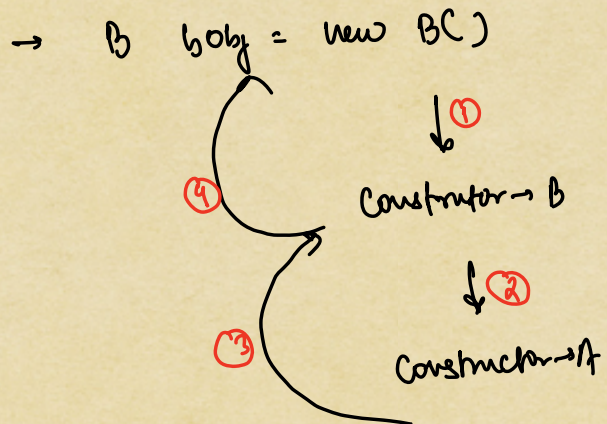
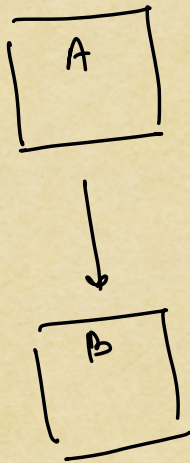




\* Object of the child class is created from child class constructor, variables (inherited) of the object will



be initialized by parent cons.



\* constructor chaining ↓

calling + constructor body  
to call of multiple cons. down  
the line



Animal

int legs

public Animal(int legs) {

}

Cat c = new Cat(4)

Cat cons.

Animal cons.

Safari2022

- music()  
- navigation()  
- climateControl()

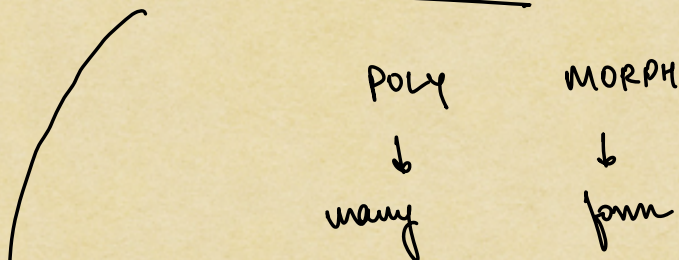
Safari2024

→ 360 camera()  
→ ventilatedseats()

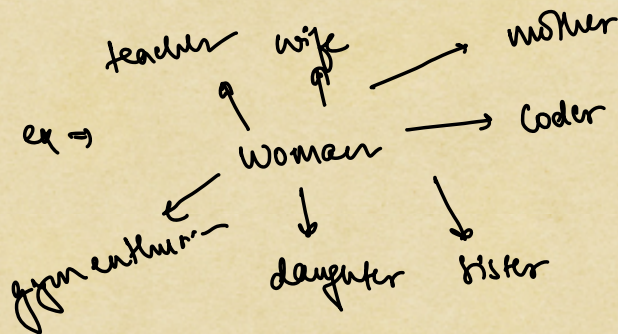


→ ADALC)

⇒ POLYMORPHISM



→ ability of something to be present in multiple forms



⇒ method overloading:

```
public int run(int speed, int dist) {
```



method signature



✓ name + args data types

⇒ [run(int, int)] ✓



```
public int eat (int roti, int paratha) {
```

```
}
```

↓  
eat(int, int)

```
public int run() {
```

```
}
```

↓  
run()

run(int, int)

In a class you can have multiple methods with same name but diff. args.

⇒ Method overriding

