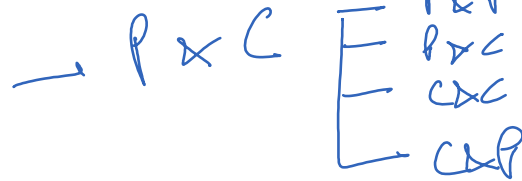
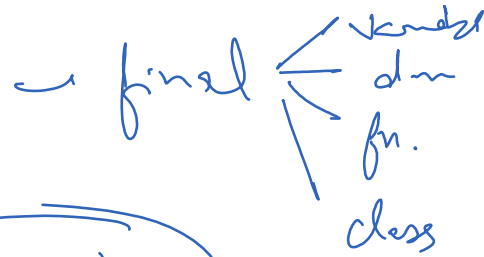




Inheritance & Polymorphism



Golden Rules.



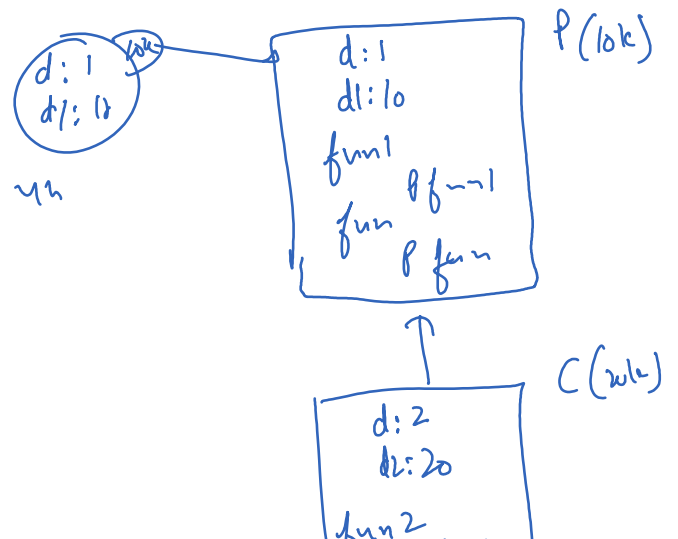
3:55 - 4:00

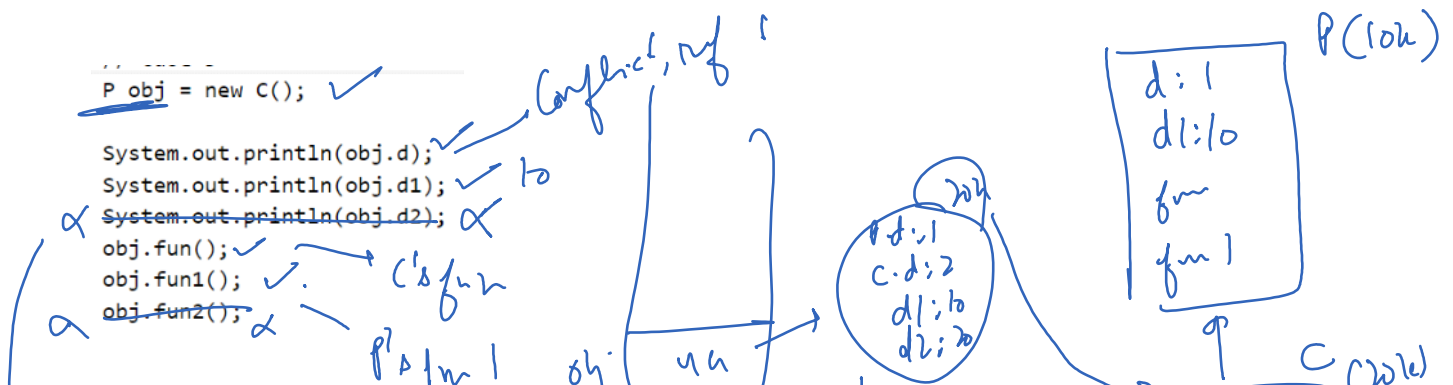
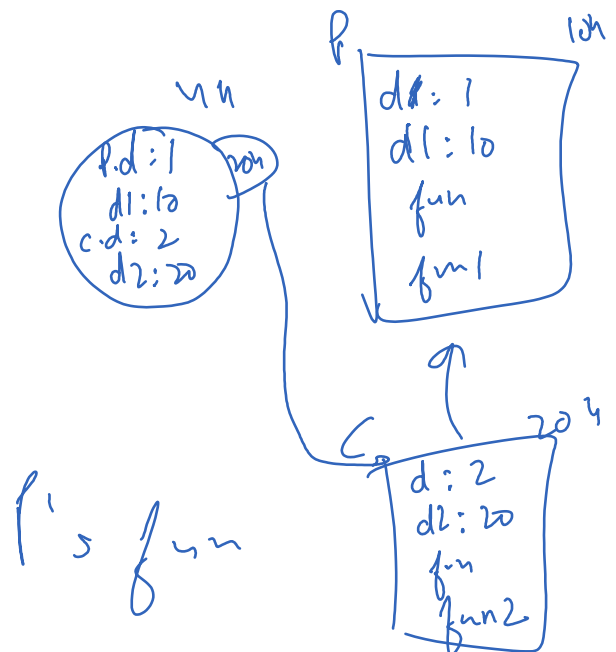
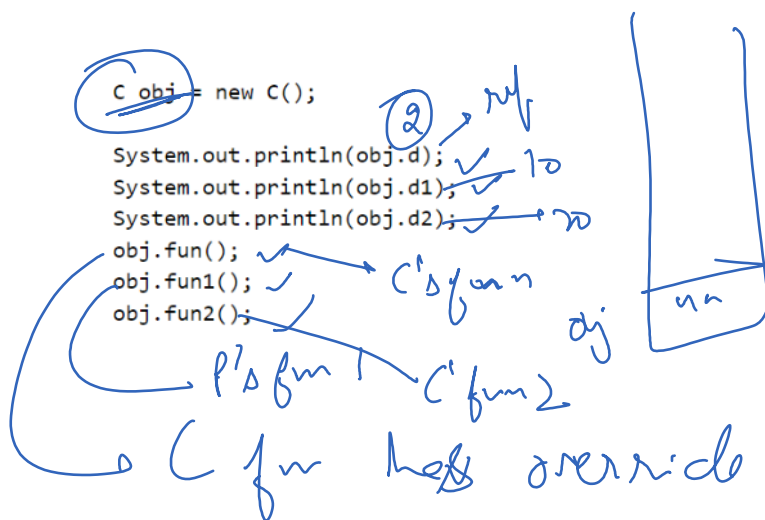
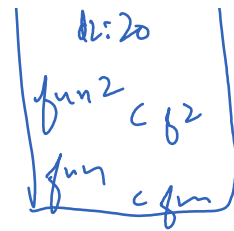
```

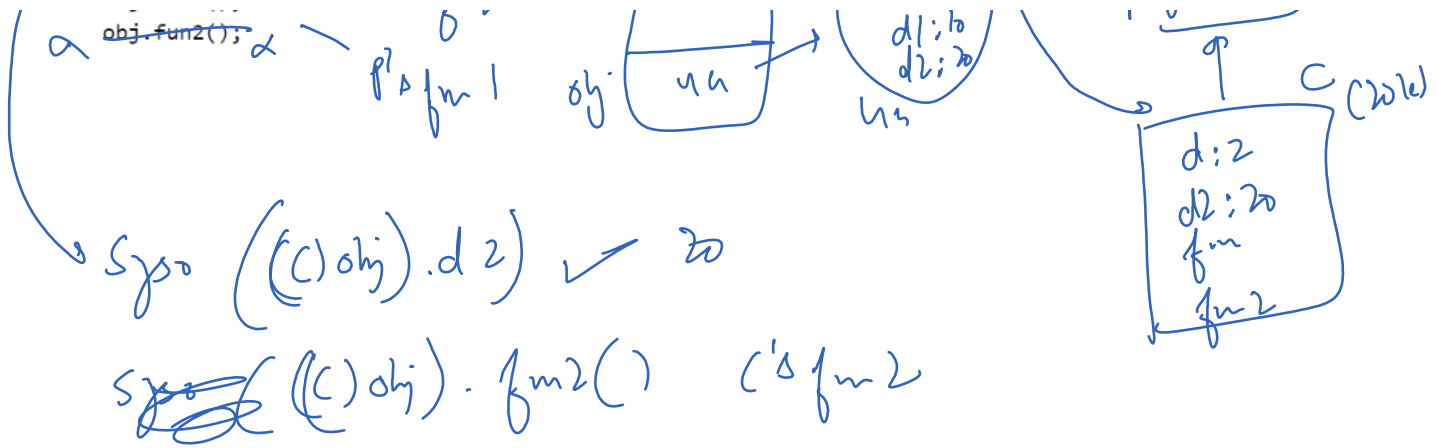
P obj = new P();
System.out.println(obj.d);
System.out.println(obj.d1);
System.out.println(obj.d2);
obj.fun();
obj.fun1();
obj.fun2();
  
```

① ✓  
② ✓

✗ → compile time  
✓ 1 fn  
✓ 1 fn  
✗ → compile time

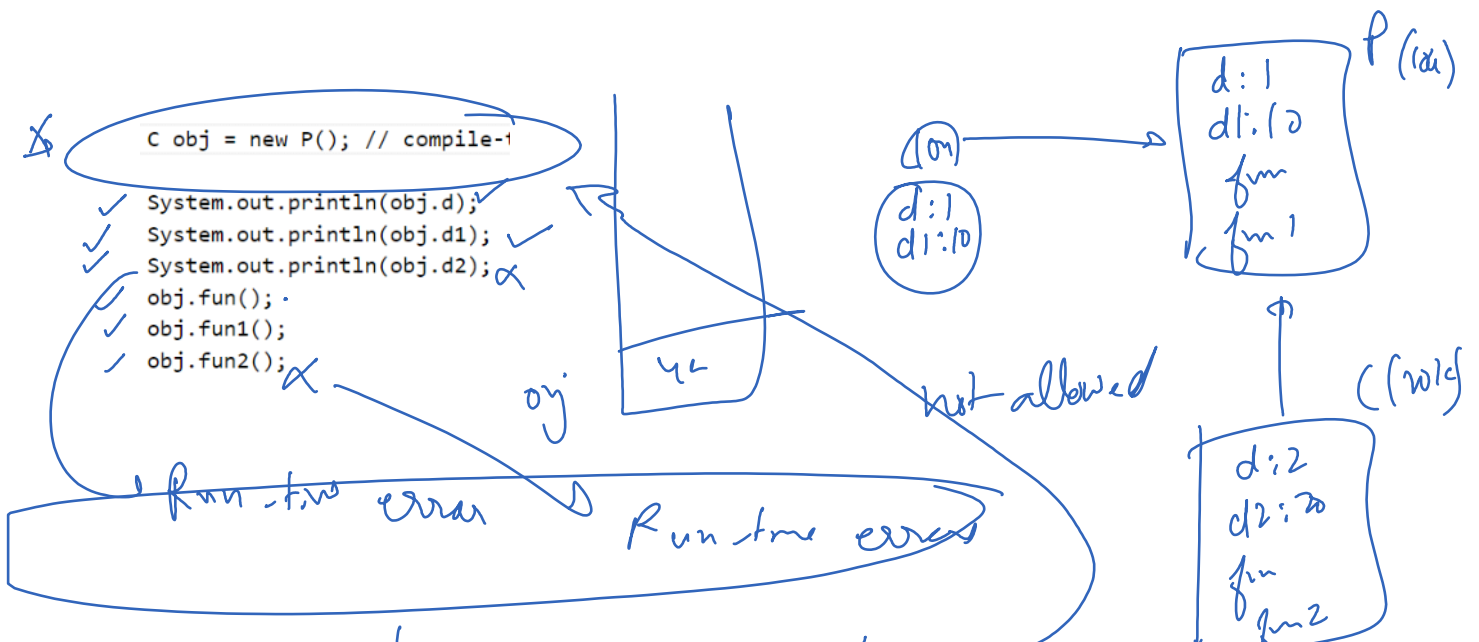






```
C obj = new P(); // compile-1
```

```
System.out.println(obj.d);
System.out.println(obj.d1);
System.out.println(obj.d2);
obj.fun();
obj.fun1();
obj.fun2();
```



fun  
fun2

— Notus

## Doubts

find

absurd  $\hookrightarrow$

BA fm 2 ✓

non-schiz Bl. für (O) X

2.2

$$b < 1$$

A hand-drawn diagram of a rectangular box. Inside the box, there is a vertical line. Below the box, there is a label '2'.

 $\sigma_f(\omega)$ 

$\text{poly} = \text{null}$

$$1/(s=1) \int_{\mathbb{R}^d}$$

Bsp

$a_i \in m$

$S_{x_i} = 5$

$s \text{ fun}$  ( $x_i$ )

$\rightarrow \text{fun } i$

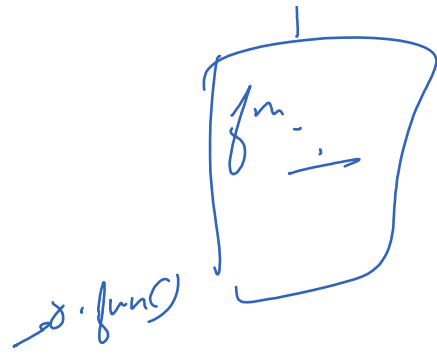
$m -$

$i$

$\text{Pres}$

$S \subset \text{Skal}$

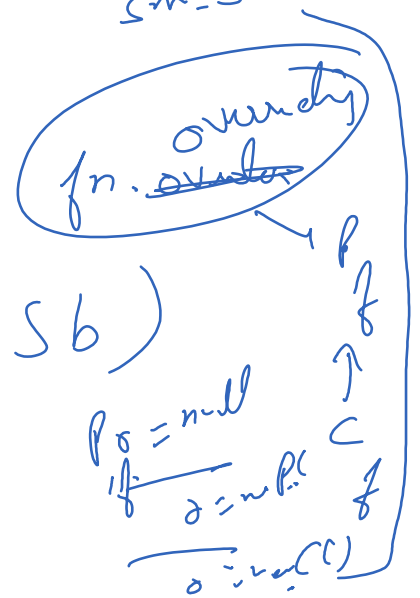
if (s == 1) '  
     o = new P();  
 else if (s == 2)  
     o = new C();



fn. overloading → compile-time polymorphism  
 sum(i a, i b) } sum-i  
 sum(l a, l b) } sum-S

sum('A', 'B')  
 sum-f("A", "B")

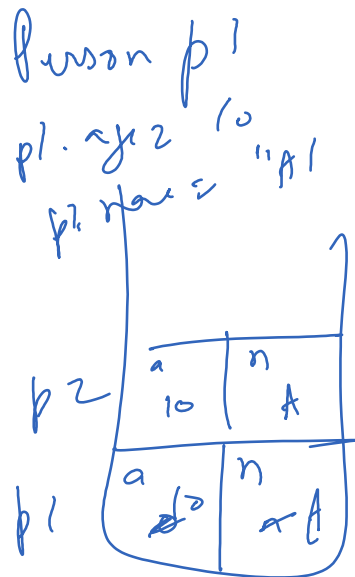
sum(S a, S b)  
 a < b



Array.sort(int[] by  
 size) arr

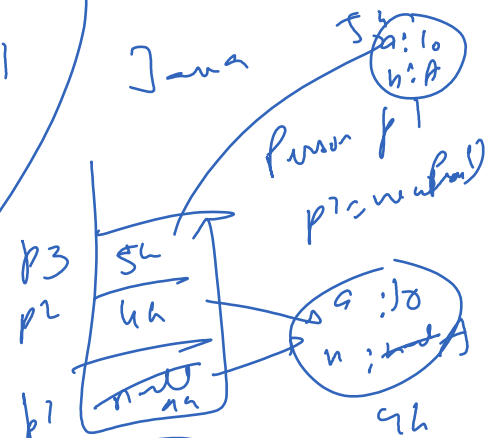
To minimize the ABI bco

C++



Person p2 = p1

Java



Person p3 = new Person(p1)

Person p2 = p1