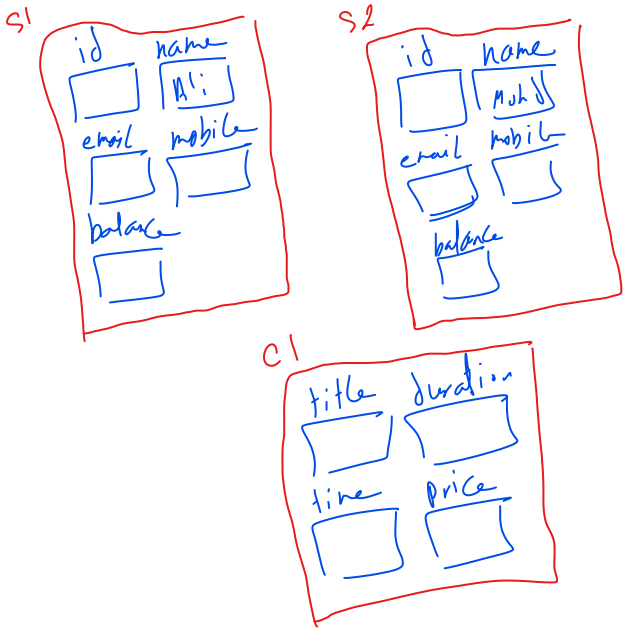



OOP

object → special variable
→ place in memory

class → code

How to create object?

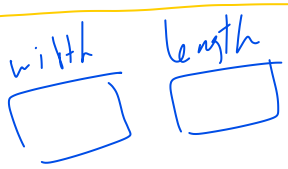
Define operations on object

Student

Trainer

Course

---



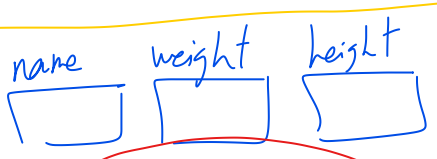

area = width * length

OOP

Room


```
__init__(w, l)
get_area()
```

---




$$bmi = \frac{weight}{(height/100)^2}$$

- <18.5 Underweight
- <25 Normal
- <30 overweight
- Obese

OOP

Patient

```
__init__(n, w, h)
get_bmi()
get_status()
```


annual_salary = salary * 12

service_period = 2022 - hire_year

## OOP

### Employee

__init__(...)

get_annual_salary()

get_service_period()

---


$$pct = \frac{mark}{full\_mark} * 100$$

- >85 Excellent
- >75 V. Good
- >65 Good
- >=50 Pass
- Fail

## OOP

### Student

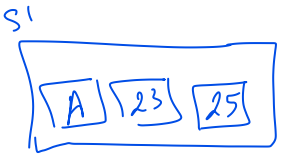
instance variables { name, mark

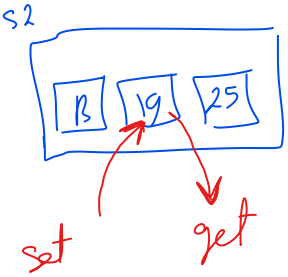
static variable ← full_mark

__init__(...)

get_percent()

get_grade()




Encapsulation

---

## Account

name

- balance

annual_rate = 0.04

__init__(...)

get_monthly_rate() → /12

get_monthly_int() → * balance

deposit(amt)

withdraw(amt)

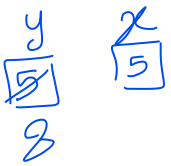
get_balance()

---

x = 5

y = x

y += 3



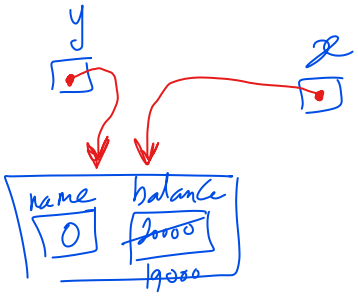

copy value

---

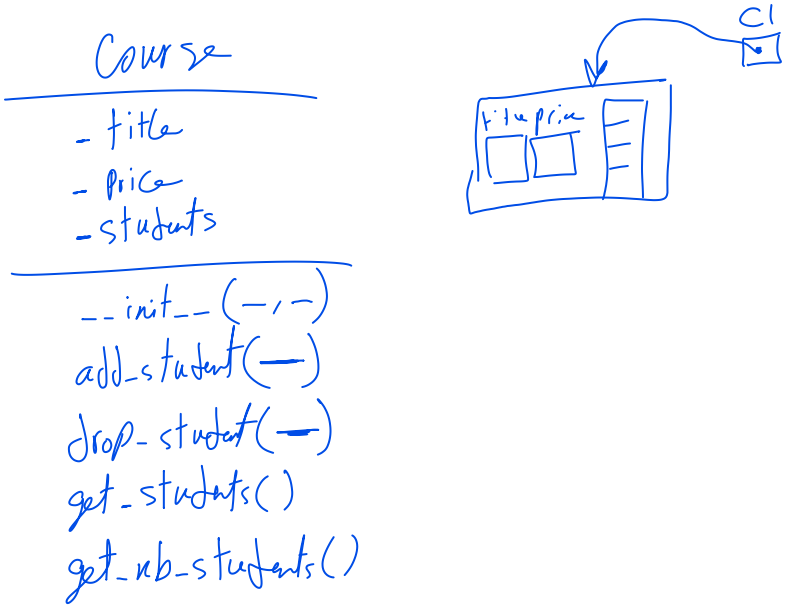
x = Account("0", 20000)

y = x

y.withdraw(1000)

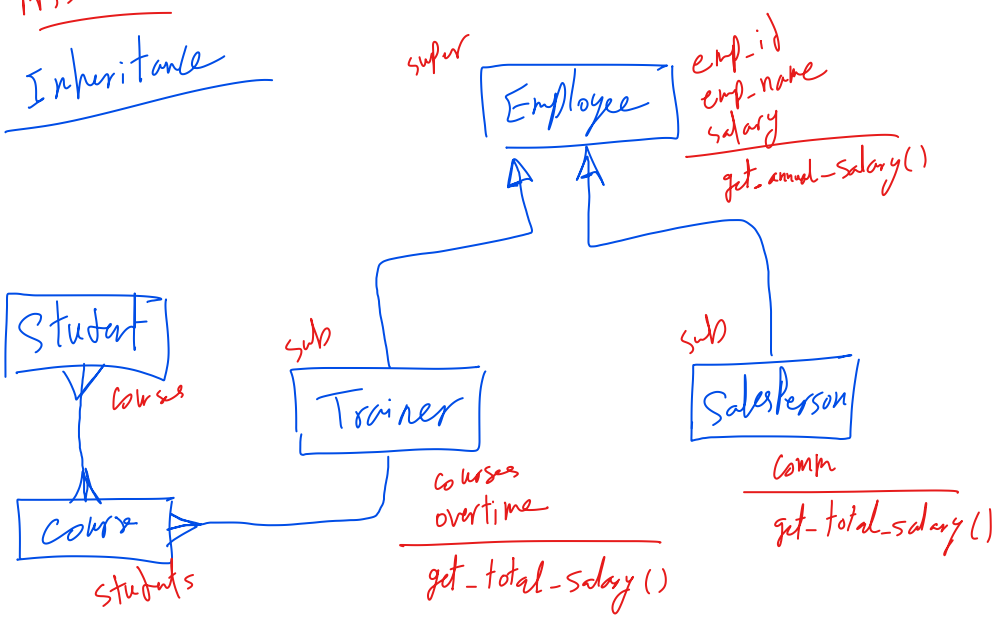



copy Reference




Association

Inheritance



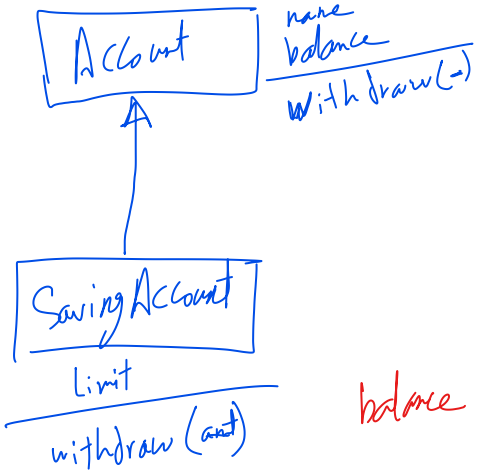

t1 = Trainer(---)

t1.

| overtime |
| --- |
| emp_id |
| emp_name |
| ⋮ |

## OOP

* Abstraction
* Encapsulation
* Inheritance
* Polymorphism




balance - amt >= limit