- Summary
- Today's content:
  - a math question
  - String vs Char Array
  - Static
  - Inheritance
  - final keyword
  - Polymorphism
  - Enums
  - Generics

- $-40 \% 9$ — Java

Q.

Divisor | Dividend

maths

Remainder = Dividend − (Div × Q)

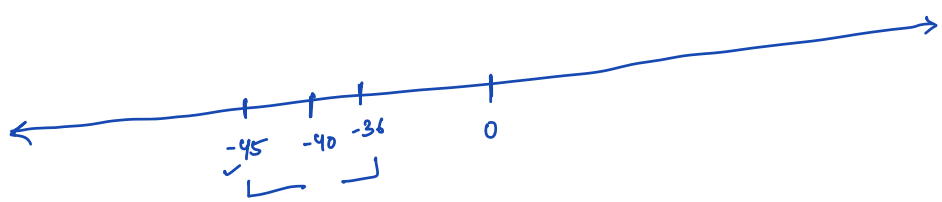
↳ closest multiple of Div <= Dividend

$124 \% 15 = 124 - 120$

$120 <= 124 = 4$

$-40 \% 9 = -40 - (-45)$

$= -40 + 45 = 5$ $<= -40$



Java

$-40 \% 9 = -40 - (\text{Div} \times Q)$

$9 \times \left(\frac{-40}{9}\right)$

↳ -4

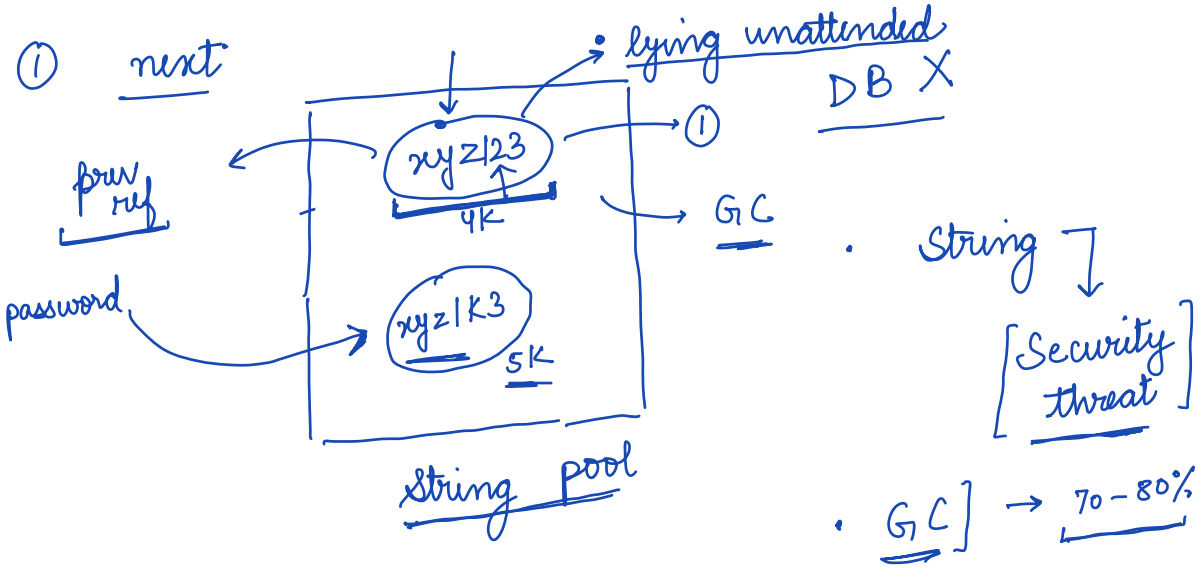
↳ -4

$\frac{-40}{9}$] = int = -4

$= -40 - (9 \times -4)$

$= -40 + 36 = -4$

① next



. String] → [Security threat]

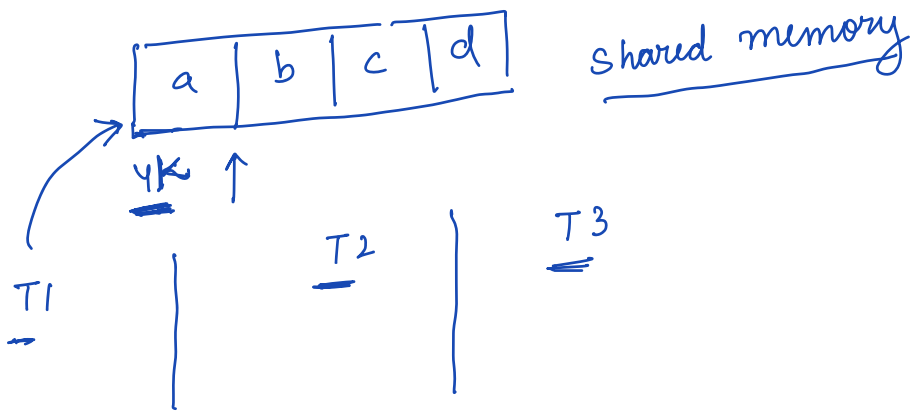
. GC] → 70-80%

② array → print [logs]

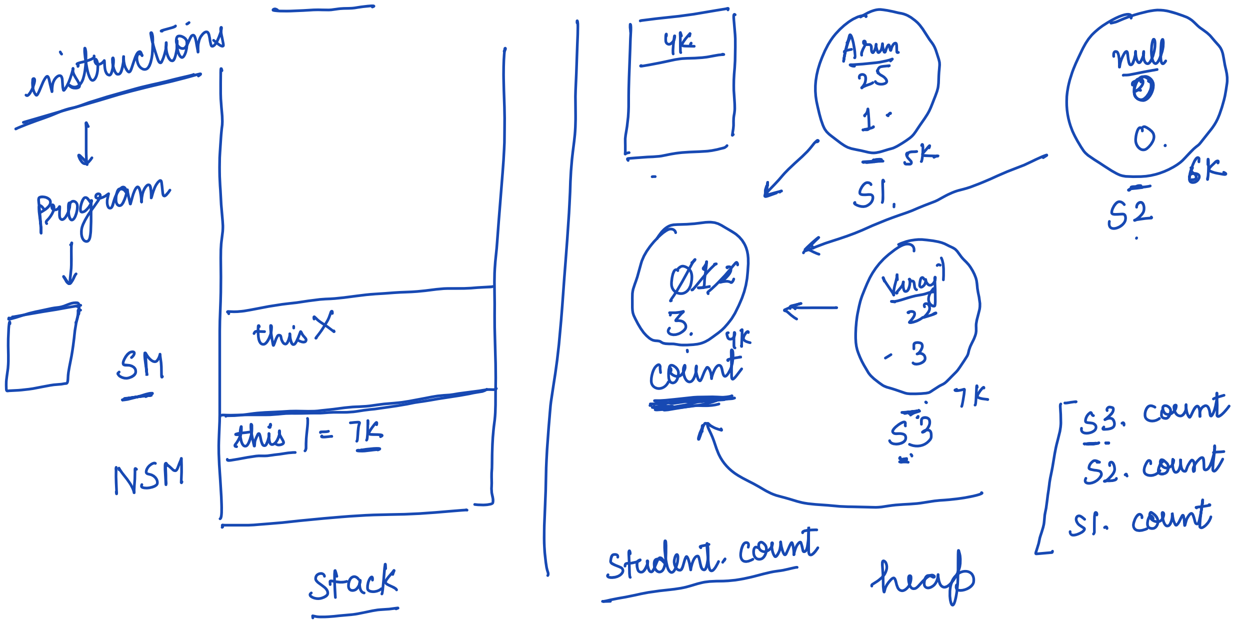
↳ password ↳ ref / add / location

String → print

↳ content.



Static – entire class – common to all objects



✓ Static function : → class p → this X

↳ Non static variables ref. X this

✓ Non Static functions → object → static variables ✓

Static class → accessible [methods, variables]

- mandatory :
  ⇒ [static functions, static variables]
  ↓
  ✓ no object

{ }

no object – limitation

↳ 1. private constructor ✓ – optional

optionals 2. final – inheritance

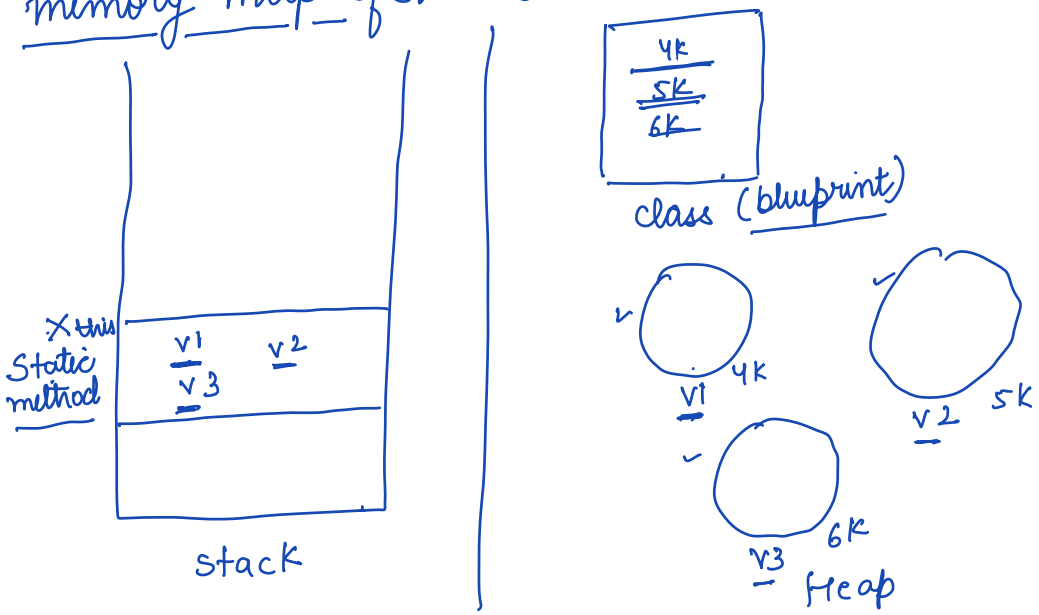
3. static functions & static variables

Usage →

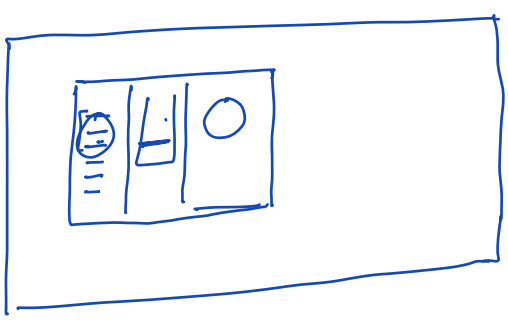
static ← [Math . sqrt] →
. max
. abs

- static class → overridden or overloading

memory map of static class :

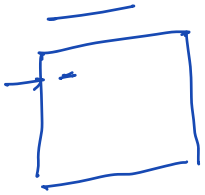


Inheritance



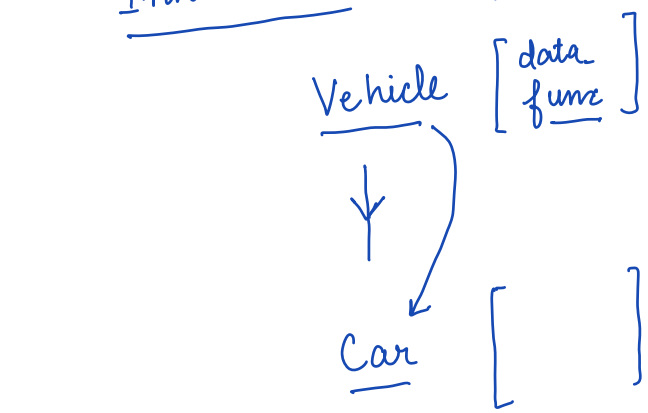
instructions

class → [≡]



break 10:40 pm

Inheritance : code re use ✓

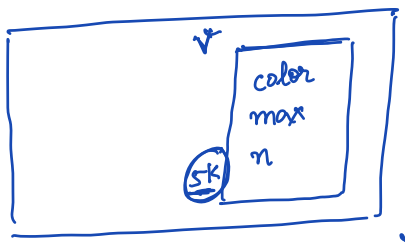


Vehicle V = new Vehicle ( )

Car C = new Car ( )

Vehicle
color
max – private
numtyres
public getmaxSpeed

Car
numG
isC



child Car → super → parent

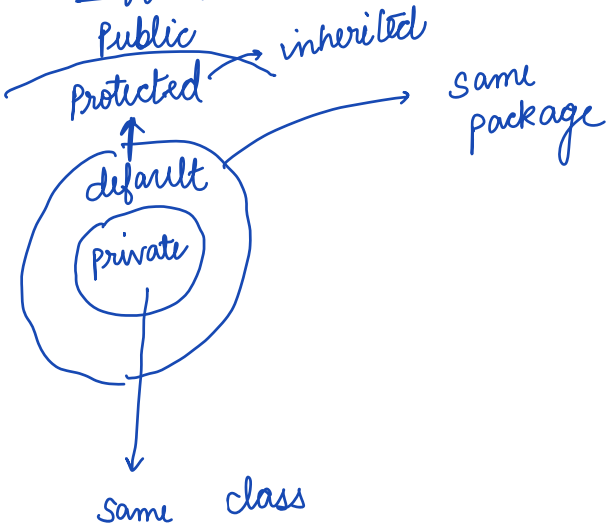
super

C = 4K

this -

Protected — same package

— diff. package, extend.



Project

Package 1

- C1
- C2
- C3

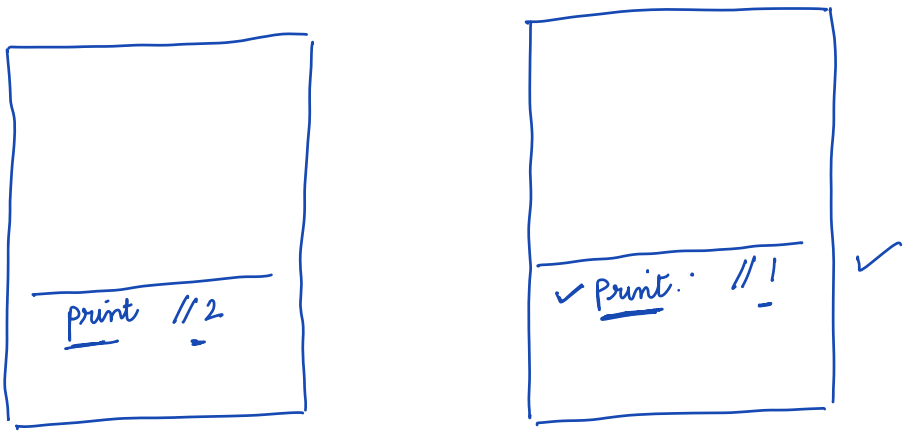
P2

- C1
- C2

P3

- C1
- C2

Polymorphism



Vehicle

Car

```
Car c = new Car()
c.print
```

```
Vehicle v = new Vehicle ();
Car     c = new Car ();
```

→ `Car c = (new Vehicle ())` invalid.

ref = c

object → vehicle

↓

Truck

Bicycle

Bike

→ `Vehicle v = new Car()`

car

Car

Enums

LOW → Int → binary

MEDIUM

HIGH

'a' → Int → binary

'b'

'c'

static final int num → update it only once