Java lect 2

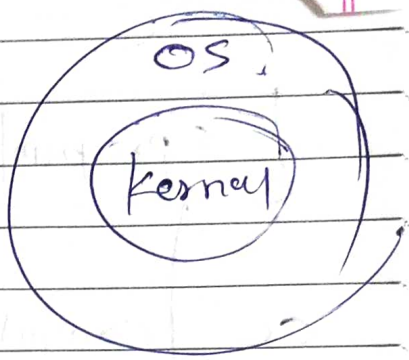
{ Atomic habits }

Page No.:
Date:

Introduction

OS

kernel + untility.



→ Unix → macOS
→ Linux
→ window

→ Unix is mother of all OS.

Android → own by Google → Larry Page
→ Sergey Brin
→ Andy Rubin.

→ Android os is written in C
→ Android applications written in java.

→ Google develop new lang flutter

iphone (ios is operating system
→ written in C, C++, objective-C, Swift, assembly language.

Swift objective C

Date:

C

C program

cc compiler

a.out

executable file

loader

OS | a.out

RAM process

C++

C++ program

compiler g++

a.out

exe file

loader

OS | a.out

RAM

process.

g++ is GNU

GNU's not unix a recursive acronym

Java

Java program

javac compiler

.class

byte code

java interpreter

JVM

binary code

OS | JVM

that is why java is slow than C & C++.

13-4-23

Java overview 01

lect 3

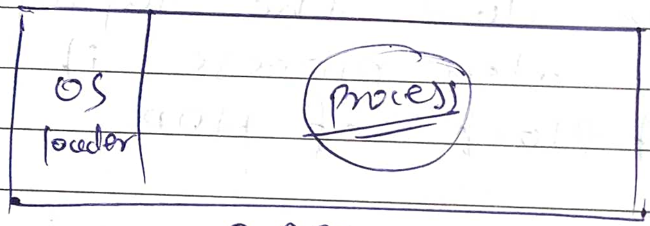
Program → it Create automated things

- Monitor → screen → 2D.
- → CPU → brain of computer
  - → access the instruction

(10+30) → 30

→ RAM → example → theatre.

↑ इथे साठवल्या गोष्टी Running मध्ये असतात.



RAM

128
+32
160
128
64

→ GPU → Graphics processing unit.

128, 64, 32, 16, 8, 4, 2, 1

65 → 1 0 1 0 0 1 0 1
170 → 1 0 1 0 1 0 1 0
20 → 1 0 1 0 1
175 → 1 0 0 1 0 1 1
4 → 1 0 0
120 → 1 1 1 1 0 0 0

64
32
96
16
112

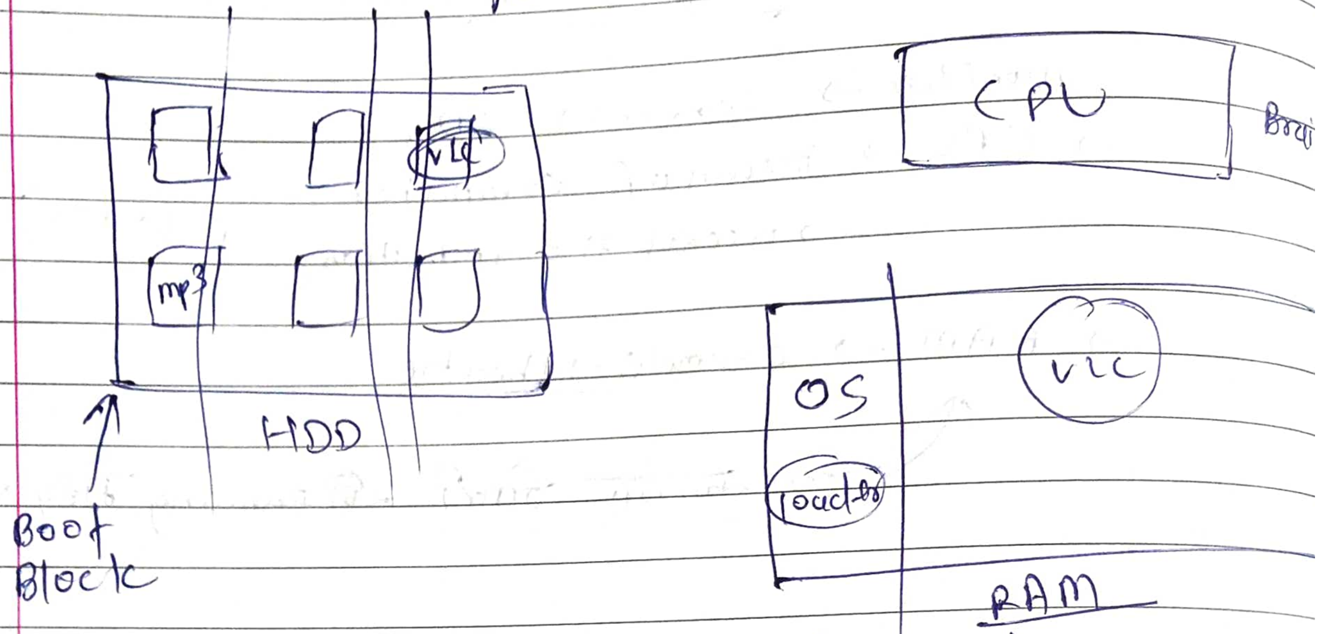
GB → 1024 × 1 MB
MB → 1024 × 1 KB
↑
KB ⇒ 1024 × 1 byte
↑
1 byte → 8 bits
↑
bit → 1

Java overview

denny smithy & ken thomson → C founders

lect 4

Ram type → DDR1, DDR2



When we close the laptop (PC) Then operating System also process it close goes to the Boot Block of HDD.

filesystem

windows OS → ntfs (new technology file system)

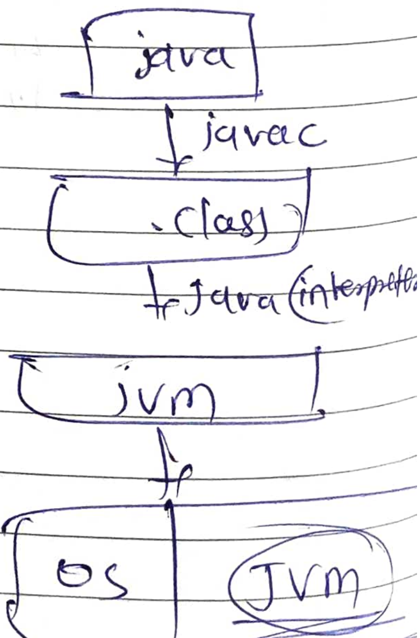
linux OS → ext3/ ext4 (extended)

maccOS → ~~APP~~ AFS → Apple File system.

GNU linux → GNU not unix.

Python interpreter → in C

JVM → in cpp.

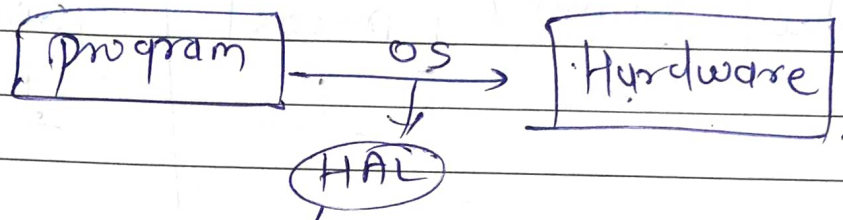


14-4-23 lect 5 Java overview 03

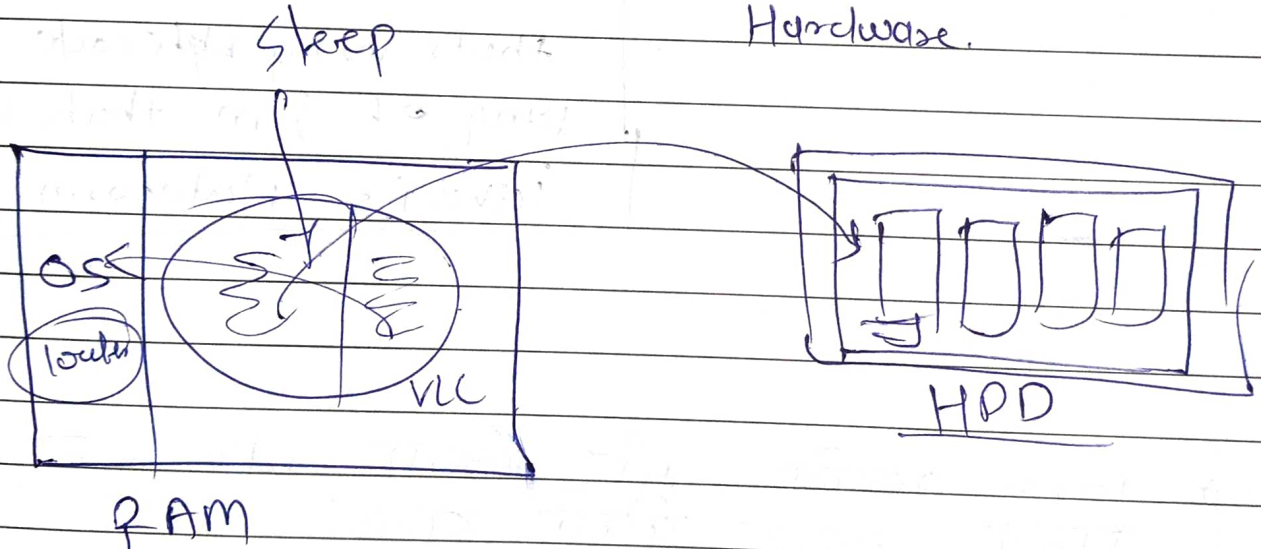
M T W T F S S
Page No.:
Date:
YOUVA

Hardware
- monitor
primary - RAM
- CPU
- mouse
- keyboard
secondary - HDD

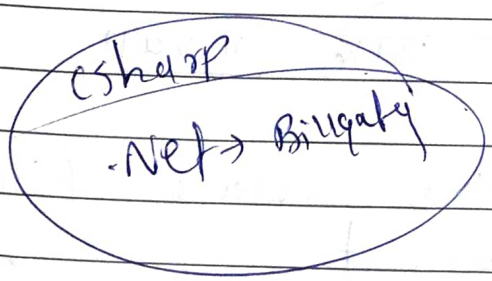
OS → incharge.

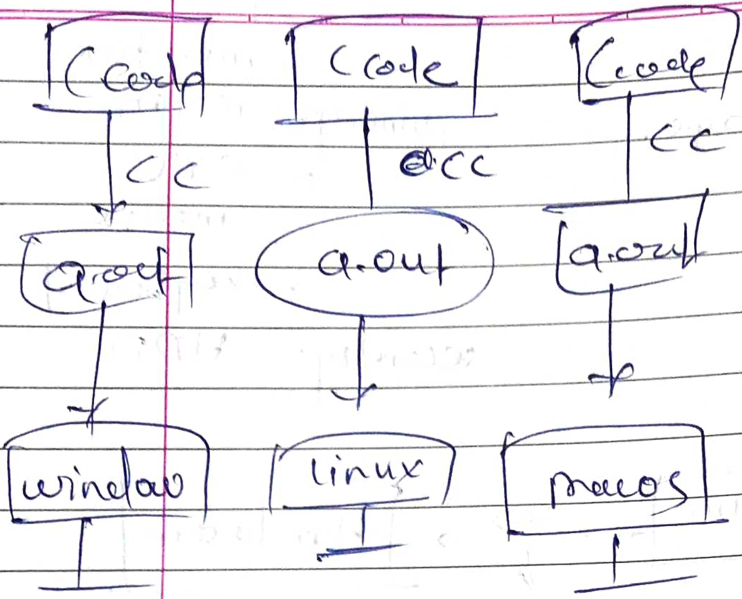


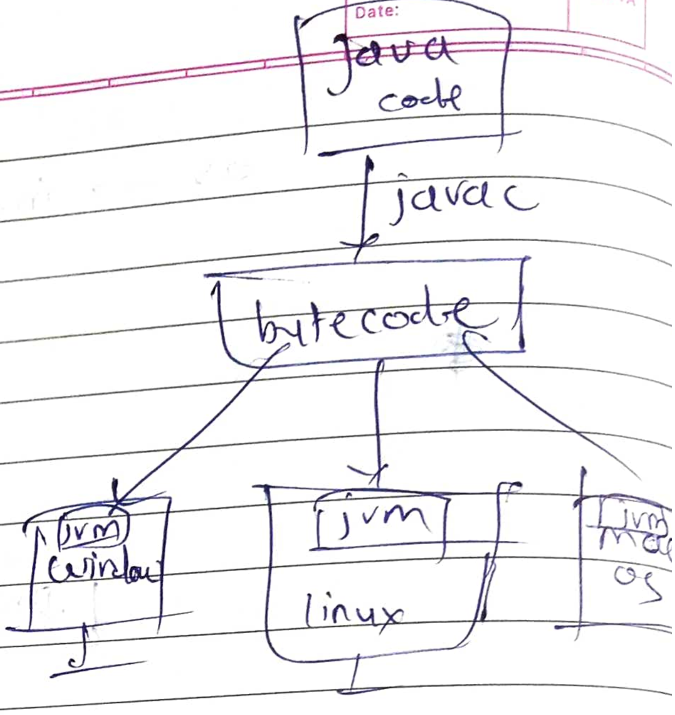
Hardware Abstraction layer
↳ for protection of Hardware.



$ ps-elf
↳ command for seeing processes.







In Java bytcode is platform independent but jvm is plat form dependent

thats why bytecode is lang of jvm thats why java is platoform independent

→ Java म्हणतो मी माझ्या jdk त्या kit मध्ये गोष्ट दिली होती ज्याने .class file उघडून चांगल्या format madhe देखू शकतो.

त्याचे नाव आहे javap

javap – open the .class file in good format

javap is part of the official java tools

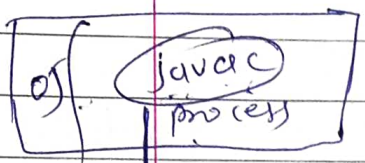
p → in this print

18-4-23

jvm- Architecture

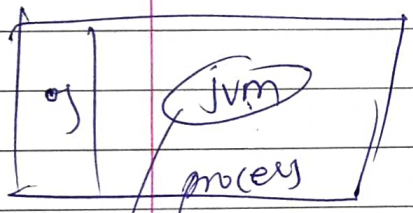
hardware keyboard

after writing code notepad process is closed

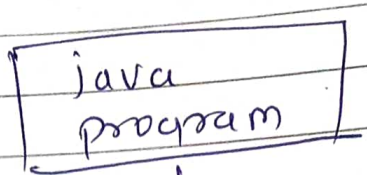


javac as process create the .class file save at side of the program file

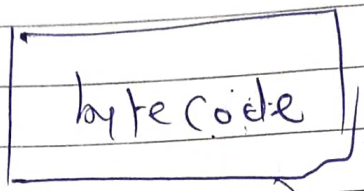
Then



take the .class file as input. with help of OS it converts the bytecode int binary code

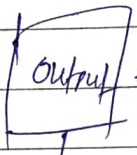


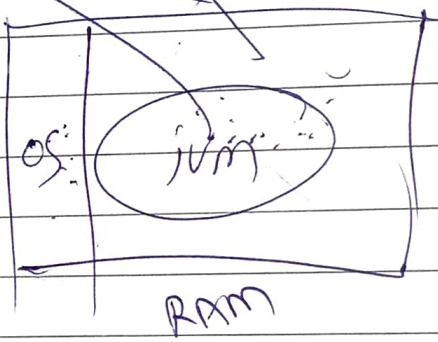
javac → compiler
- convert into byte
- check the syntax

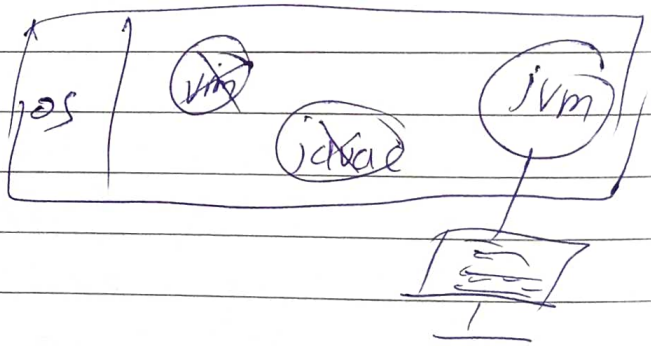


→ platform independent

.class







ls → list files

PS -elf → for seeing processes

bash → terminal

:wq → save quit from editor

→ for compilation & javac program1.java

javap -c Core2Web.class
- open bytecode

→ java filename → for run

java is virus free lang
↳ ~~burst~~ robust

jvm is ~~चोरसंवढ~~ → सर्व गोष्टी वाटतात देखील.
in bytecode

java is pure object oriented lang.

→ vim core2web.class → opening bytecode
दुसऱ्या editor मध्ये bytecode open करू नका.

for delete (rm core2web.class).

→ magic number unique ~~ast~~ असतो प्रत्येक file ला.

→ java program2.java → we can directly use this command for run the program.

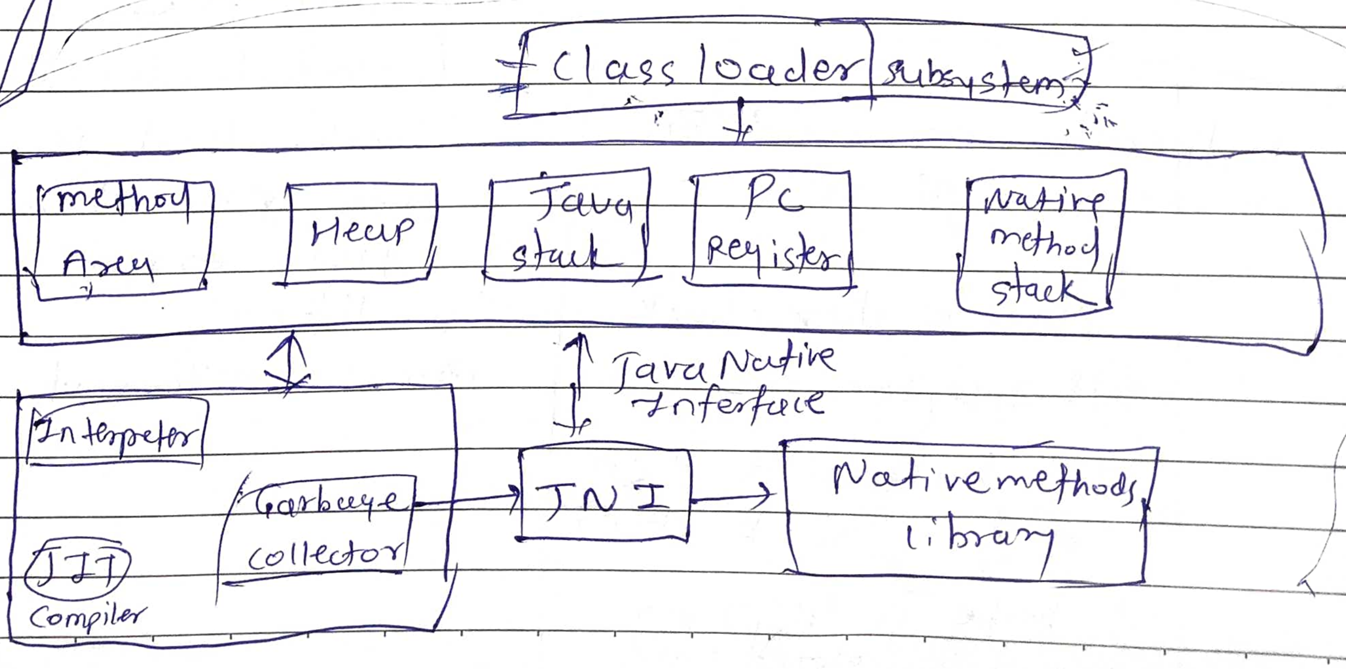
↳ java ~~is~~ पण python सारखं करतो

JVM 😊 → मोठ्या गाली byte code generate करून आणि run करेल

→
marshmelo
orieo
kitkat
nogat
lollypop
icecream
sandwich

80% - 85%
→ 1.8 java version using.

before 1.7 jvm



lect7

19-4-23

jvm architecture

java overview:

Java 1.6) → पर्यंत main नसला तरी रन व्हायचा code.

1.8 ← more stable version (most used in company)

1.11 ← stable

1.17 ← LTS (long term support).

java first version → (1995) → Sun Microsystem
→ James Gosling
→ 2008
→ oracle

bash → bowrne again shell

मराठी माणूस unicorn

Quickhill → Virus Company.

NPAV

→ classloader subsystem → protect jvm

it check the .class file

↳ classloader subsystem check if it is not format for jvm then it will gives error like

error: class format error

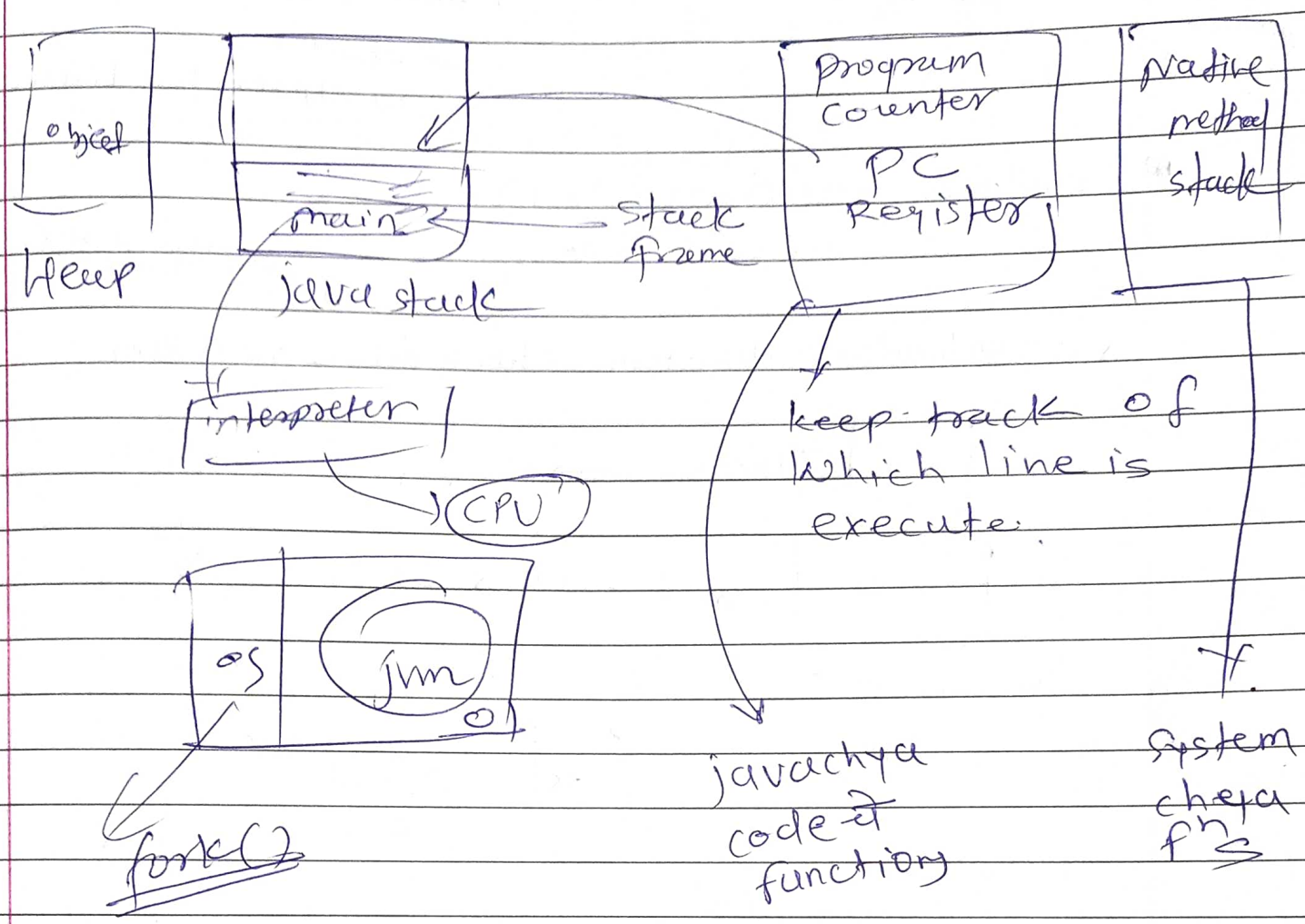
bytecode → root ओळखतं

↗

classloader check this bytecode

कसं bytecode ओळखतं? check करतो.

→ byte code proper आहे का?



Class loader subsystem : byte code verifies & check versions if right puts into method area

Method area : used to store bytecode & manythings.

Heap : classes -> objects

Java stack : main fun^n & other fun^n

PC register : to track on Java stack

Native method stack : Native fun^n runs usability

Execution engine - bytecode to binary & take to CPU

Interpreter -> Line & Line

JIT : Smart compilation

Garbage Collector - Heap section unused object clears

JNI : Java fun^{n}lity & Native fun^{n}lity binding

↖ cpp code code on java

Devops fields
developer operation

lecture java (jvm) overview.

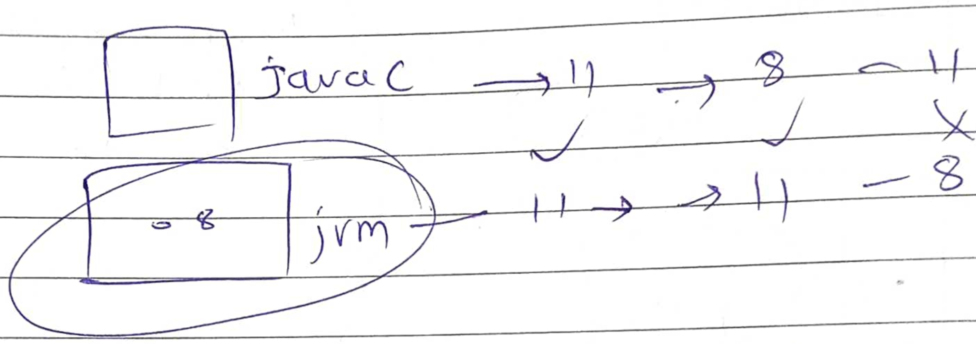
20-4-23

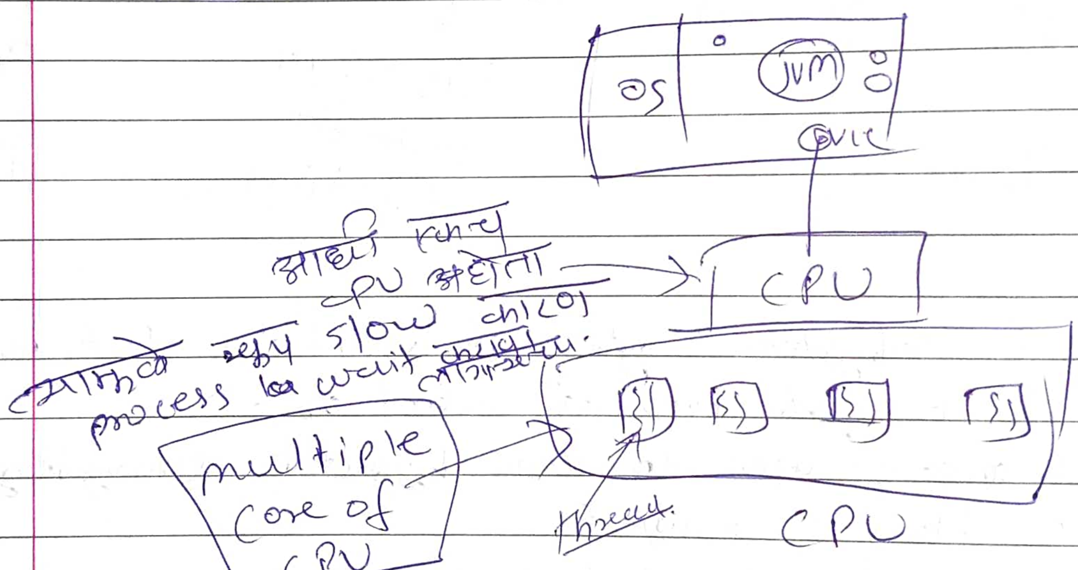
eclips IDE

install (WSL)
(windows subsystem for linux)

* windows os (book: charls petzol author) (win 32 k)

→ classloader subsystem check also versions. → verify bytecode





आत्ताचे आताच्या मशीन fast आहेत.

→ all process should want CPU

→ How many core in your computer.

example (4-core 8-thread)

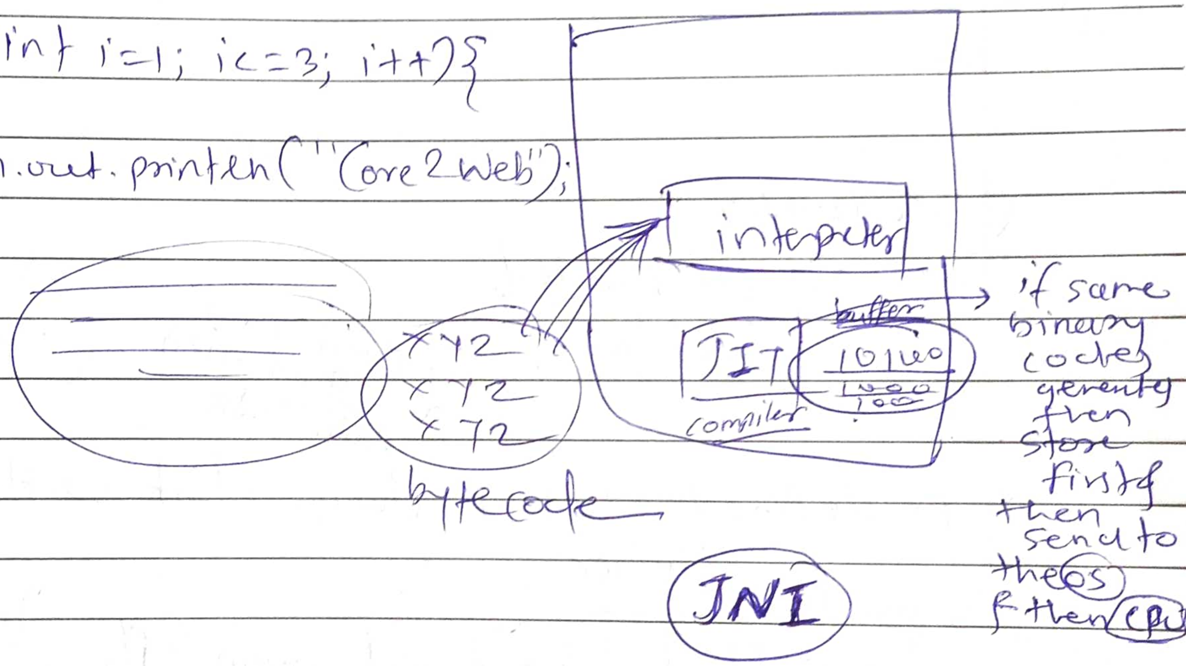
→ काही processes heavy असतात. process ना हे करतात

→ CPU shedulars OS चा भाग

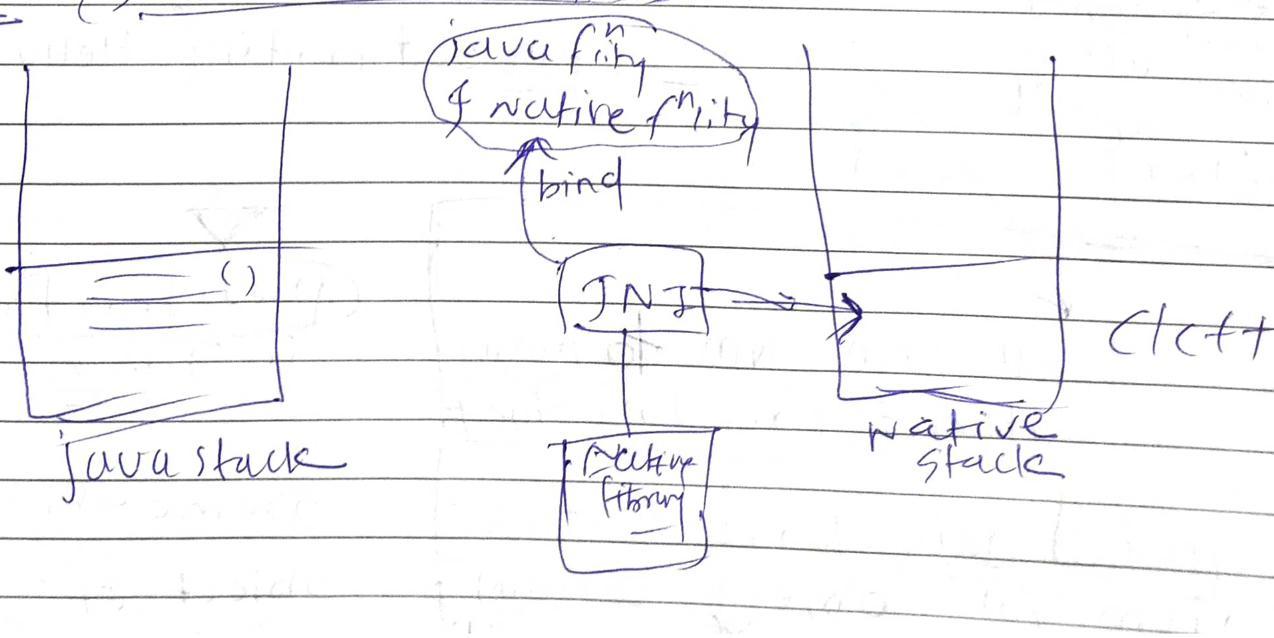
→ **JIT** → It is part of execution engine in JVM.

→ just in time
it stores looped code outputs
like for loop

```
for(int i=1; i<=3; i++){
System.out.println("Core2Web");
}
```



* **JNI** (Java native interface)



oak → java

entry point function

class demo {

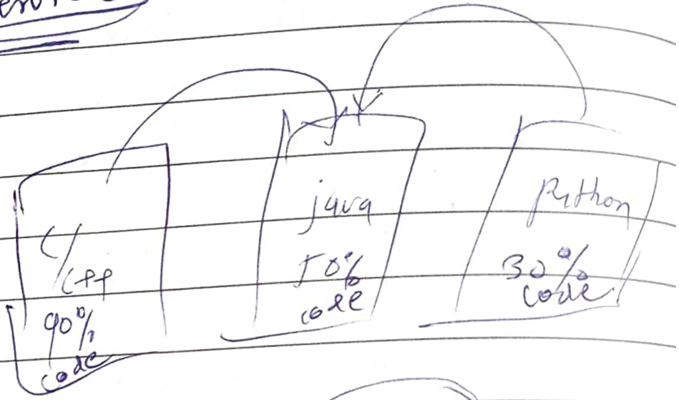
Public static void main (string[], args) {

Shark tank

lect 9 Main method in Java

Classes in java → Jar → runtime → java archive (Jar: java archive)

jems glossing Interview



Zerodha — self funded, bootstrap

ZOHO — CEO श्रीधर वेंबु

डाइरेक्ट call करना होता है (public main)

→ predefined classes always start with Capital letter

class demo {
Public static void main (String[] args)
System.out.println("Hello");

userdefined — These is not mandeltory of capital letter

String, System → Predefined classes in Java

call from jvm to main without creating object

static नहीं write करते तो JVM को object बनाना पड़ता examples → Demo.main()

Demo obj = new Demo

object ही key आदि (चाबी)

lect 10

Keyboard
↳ Non printable
↳ printable

Datatypes 01

* Naming conversion.

demo
class Demo{
}
this also convention

1. class
   ↳ capital letter → it is not compulsory

class Demo{

only two things in class → 1. variable / 2. methods / functions

}

Run Rate
16 run in 2 over (per over)
$\frac{16}{2} = 8$

→ sizeof fn java मध्ये नाही.

* Datatypes

4 byte int
4 byte float
2 byte char
8 byte double
8 byte long
byte
short
boolean

runs = 180
height = 5.6
most IMP

void ~~म्हणून~~ datatype नसून आहे but java मध्ये नाही सांगितलं तो त्याला Return type आहे.

~~jvm~~ java smart आहे तो memory manage करतो आहे ना.

→ 1 byte
(?) 1 byte
(char)

In java
char grade = 'अ';

Java by default double ला support करतो.

A-Z → 65-90

a-z → 97-122

java 2 byte char

2-unicode

English | language marathi

JVM मध्ये threads असतात.

JVM is mini operating System.

Size वाढायची असेल तर wrapper class picture मध्ये येतो.

```
class Demo {
    in main( ) {
        int a;  sop(a);
    }
}
```

error: variable a might not have been initialized.

25-4-23

Page No.: 84 2

Date:

Datatypes 02

error: cannot find symbol

jerNO = 18;

Symbol : variable

location : class Demo

```
public static void main(String args[])
    byte var1 = 18;
    byte var2 = 18;
    System.out.println(var1);
    System.out.println(var2);

    var1 = var1 + var2;
    System.out.println(var1);
    System.out.println(var2);
```

or ~~Sysl~~

output:

18
18
36
18

} not generate output.

this error also for using short datatype

error : incompatible types : possible lossy conversion from int to byte

var1 = var1 + var2;
^

java ची runtime value generate होते तिला integer असं consider करतो.

float को java ignore करके उसे float value को double consider करना।

```
class Demo{

    public static void main(String[] args){

        float f1 = 7.5;
        float f2 = 7.5;   → by default consider double
        (4 byte आहे)     (8 byte चे आहे)
        System.out.println(f1);
        System.out.println(f2);

}}
```

output:

7.5
7.5 } not generating output.

java:6: error: incompatible types: possible lossy conversion from double to float
float f1 = 7.5;
^
double float मध्ये convert करायचा प्रयत्न करतो

java:7: error: same as above
float f2 = 7.5;
^

because 2 variables ← 2 errors.

write code for above is:

Solution:→
float f1 = 7.5f;
float f2 = 7.5f

byte integer चे store करतो.
datatype

compatible ~~err~~ type error → चे example
आईने कुंद्य आनायंमो लपवो
आपण छोट मोठं बेऊन जो (वाटी)

System.out.println (var1 + var2)

print the value without storing.
(direct त्याला कुंद्य ओतणे
store न करना).

→ java मध्ये class च्या बाहेर काहीच चालत नाही.

→ java मध्ये global variable नावाची concept च नाही. त्याच्या जागी instance variable.

```
class CricPlayer {
        int x = 10;    // instance variable
        public static void main(String[] args) {
                int y = 20;  local variable.
                        ( java stack वर जाणार )
                System.out.println(x)
                System.out.println(y)
        }
}
```

heap section ला जातो

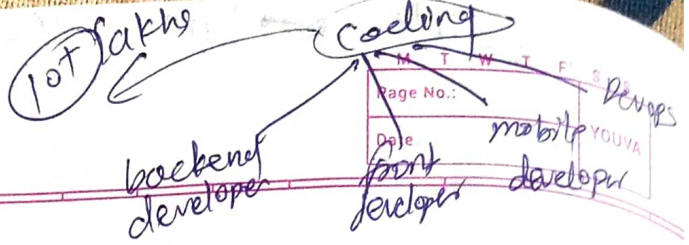
output :

error: non-static (method or variable) (x) cannot be referenced from a static context.
System.out.println(x);
                    ^
1 error

→ x हा main च्या बाहेरचा आहे त्यामुळे तो एकतर static असणं पाहिजे नाहीतर त्याचा object बनवून त्याला access करणं शक्य.

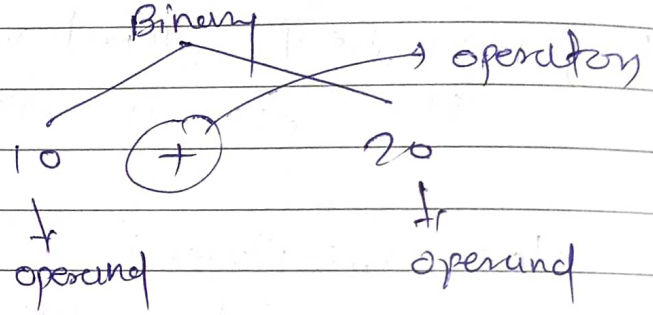
26-9-23

Datatypes 03



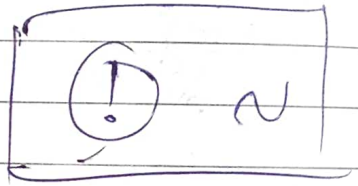
* Operators in Java

1. Arithmetic (+, -, /, %, *)
2. Relational (<, >, <=, >=, !=, ==)
3. Unary (-, +, ++, --)



4. Logical (&&, ||, !)

GHAR topic आहेत सगळे.



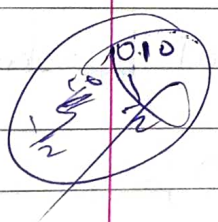
Special operators

5. Bitwise (&, |, ^, <<, >>, >>>)

6. Ternary

* code1 → Arithmetic operators.

```
public static void main(String[] args){
    int x = 10;
    int y = 20;
    sout(x+y);   30
    sout(x-y);   -10
    sout(x*y);   200
    sout(x/y);   0
    sout(x%y);   10
```



② public static void main (String[] args){
int x = 4;
int y = 5;
int z = 6;
int ans = x+ y*z + (x-y);

4+ 5*6 + -1
4+ 30 -1 => 33

System.out.println(ans); 33

Codes **Relational operators.**

```
int x = 10;
int y = 20;

System.out.println(x < y);   true
System.out.println(x>y);     false
System.out.println(x<=y);    true
System.out.println(x>=y);    false
System.out.println( x ==y);  false
System.out.println (x != y); true
}

if (x<y)
    System.out.println("Hello");
else
    System.out.println("Hiiii");
```

→ Relational operator 2 का output कैसे देगा Return true or false.

Page No.: Date:

for C
compile → cc program3.c
run → ./a.out

(2)

1 → True
0 → False

```
int x = 10;
int y = 20;

if (x)
    System.out.println("Hello");
else
    System.out.println("Hiiii");
```

in C this is valid. कोई भी number को true consider करता है।

output:

error: incompatible types: int cannot be converted to boolean

```
if (x)
    ^
```

1 error.

if को तो boolean value चाहिए।

Unary operator code (1)

```
public static void main(String[] args) {
    int x = 5;
    int y = 7;

    System.out.println(++x);    6
    System.out.println(++y);    8
    System.out.println(--x);    5
    System.out.println(--y);    7
    System.out.println(x);      5
    System.out.println(y);      7
```

++x → x = x+1 ; --x → x = x-1

++y → y = y+1 ; --y → y = y-1

internally अस्स जाता

(2)

```
int x = 5;    15;
int y = 7;    71

System.out.println(x++);     5    15
System.out.println(y++);     7    71
SOP(x--);                         16
SOP(y--);                         72
System.out.println(x);       6    15
System.out.println(y);       8    71
```

(3)

```
int x = 8;
int y = 5;                  9 + 9
int ans1 = ++x + x++;       18
int ans2 = --y + y--;       8

System.out.println(ans1);       18
System.out.println(ans2);       8

System.out.println(x);      ~~9~~ 10
System.out.println(y);      ~~5~~ 3
```

x: ~~8~~ ~~9~~ 10

y: ~~5~~ ~~4~~ 3

ans1: 9+9 = 18

ans2: 4+4 = 8

27-4-23

operators02

using keywords as variables.

```
{
    int for = 10;

}
```

```
int enum = 10;
```

o/p:
error: as of release 'enum' is a keyword, and may be used as identifier.

output:
5 errors are occured because of for

```
{
    int interface = 10;

}
```

error: ~~expected~~ not a statement
int interface = 10;

error: esr ';' expected
int interface = 10;

: <identifier> expected
int interface = 10;

error = reached end of file while parsing }

```
class {
}
```

class name is required

error: <identifier> expected

main java मे दिलेला शब्द नाही
ते fn आहे.

inner class ला नाव दिल नाही तर चालत.

class Unary {

Public static void main(String[] args){

int x = 10;

This also class → Unary obj = new Unary(){

};

after compiling

→ ls

→ program1.java ~~Compile~~

↑ Unary$1.class, Unary.class

⇒ Android inner class हा concept वर dependent आहे

class Unary {

public static void main(String[] args) {

int x = 10;

int y = 20;

int ans = ++x + y++ + ++x + 1;

sop(x); // 12

sop(y); // 21

sop(ans); // 42

~~10~~ | ~~11~~ 12
x

~~20~~ | 21
y

Date: YOUVA

* logical operators * (&&, ||)

```
class logical {
    public static void main(String[] args) {
        int x = 5;
        int y = 7;
        int ans = x && y;
        System.out.println(ans);
    }
```

output:
error: bad operand types for binary operator '&&'
int ans = x && y;
^
first type: int
second type: int.
~~error~~

```
int x = 5;
int y = 7;
int ans = x<y && y>x;
System.out.println(ans);
```

true && true

error: incompatible types
boolean cannot be converted to int.

```
int x = 5;
int y = 5;
boolean ans1 = x<y && y<x;
boolean ans2 = x<y || y<x;
sop(ans1)
sop(ans2).
```

~~System.out.println(x<y && y>x)~~

~~true && true → true~~
~~output: true~~

T && F → F
F && F → F

but in C x=1, y=2
int ans = x && y;
printf(ans)

output = 1 → True

Page No.:
Date:

zero filled right shift

Bitwise operator (&, |, ^, <<, >>, >>>, ~)
| → or
~ → compliment

$2^{31} \ldots 2^{11}\ 2^{10}\ 2^{9}\ 2^{8}\ 2^{7}\ 2^{6}\ 2^{5}\ 2^{4}\ 2^{3}\ 2^{2}\ 2^{1}\ 2^{0}$

128 64 32 16 8 4 2 1

x=5 → 1 0 1

y=7 → 1 1 1

1 1 0 1 1 0 → x = 54

		&			|
0	0	0		0	0	0
0	1	0		0	1	1
1	0	0		1	0	1
1	1	1		1	1	1

x=9
y=14
SOP(x&y);
SOP(x|y);

8 4 2 1
x=9 ⇒ 1 0 0 1
y=14 ⇒ 1 1 1 0 (&)
1 0 0 0

(|) or
1 0 0 1
1 1 1 0
1 1 1 1

8 & 15

XOR

0	0	0
0	1	1
1	0	1
1	1	0

int x = 8;
int y = 10;
SOP(x&y); 8
SOP(x|y); 10
SOP(x^y); 2

1 0 0 0
1 0 1 0
(&) 1 0 0 0
(|) 1 0 1 0
(^) 0 0 1 0

Page No.:
Date:

int x = 8;
int y = 10;
Sop(x<<2); 32 → left shift
Sop(y>>2); 2 → y=10

64 32 16 8 4 2 1
x = 8 ⇒ 0 0 1 0 0 0
1 0 0 0 0 0 ⇒ 32

y=10 0 0 0 1 0 1 0
y>>2 0 0 0 1 0
2 ⇒ 2

int x = 132;
int y = 75;

sop(x >> 5) 4 right shift
sop (y << 2) 300 left shift

256 128 64 32 16 8 4 2 1
1 0 0 0 0 1 0 0
after x>>5 → 1 0 0

1 0 0 1 0 1 1
after y<<2 1 0 0 1 0 1 1 0 0

256
32
+12
300

Date:

How to store -ve values.

steps:

if x = -5

1. positive 5 binary values.
2. 1's Complement.
3. 2's Complement

2^{31} 0 0 0 0 1 0 1

↑ signbit

if signbit 0 → positive
1 → -ve

signbit ← 0 0 0 0...0 0 0 1 0 1

1 1 1 1...1 1 1 0 1 0 — 1's complement

\+ 0 0 0 0...0 0 0 0 0 1

signbit → 1 1 1 1...1 1 1 0 1 1

-5

x = -7 → ~6

0 ... 0 0 0 1 1 1

1 ... 1 1 1 0 0 0 — 1's complement

\+ 0 0 0 0 0 1

1 ... 1 1 1 0 0 1 — 2's complement

x = -3 → ~2

0 --- 0 0 0 0 1 1

1 --- 1 1 1 1 0 0 — 1's Complement

\+ 0 --- 0 0 0 0 0 1

1 --- 1 1 1 1 0 1 — 2's Complement

195
128
067
3

M T W T F S S
Page No.:
Date:
YOUVA

-195

256 128 64 32 16 8 4 2 1

0 0 1 1 0 0 0 0 1 1

1 1 1 1 0 0 1 1 1 1 0 0 ·complement

0 0 0 0 0 0 0 1

1 1 1 1 0 0 1 1 1 1 0 1

```
int x = 7
sop(x>>2); // 1
sop(x>>>2); // 1

int x = -7
sop(x>>2);
sop(x>>>2);
```

0 0 0 0 0 0 1 1

1 1 1 1 1 0 0 0 1's complement

+ 0 0 0 0 0 0 0 1

1 1 1 1 1 0 0 1

1 1 1 1 1 1 1 1 0

0 0 0 0 0 0 0 1

0 0 0 0 0 0 1 0

0 0 0 0 0 0 1 1

0 0 0 0 1 0 0 0

1 1 1 1 0 1 1 1

1 1 1 1 1 0 0 0

1 0 0 0

→ Raj shamani
→ Amit jain
→ Piyush Bansal

podcast

Ternary {

```
int x = 10;
int y = 20;
sop((x < y) ? x : y);
```

true / false

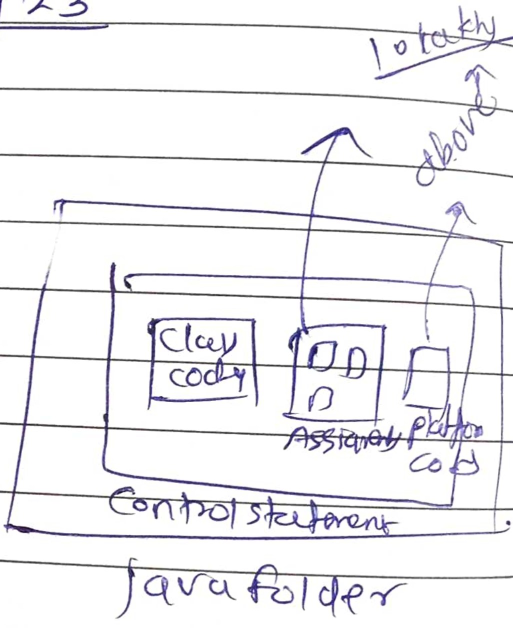
operand 1 — operand 2 — operand 3

if सारखं आहे.

lect ifelse statement :→ 28/04/23

Control statement.

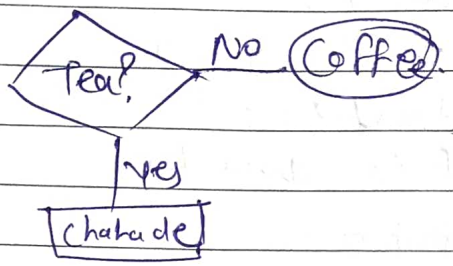
1. if
2. if-else
3. for
4. While
5. nested for & while
6. do-while
7. switch.



october पासून intership चे form निघतात google & microsoft सारख्या companies चे.

if & if-else

Q.1 Given



Q. Given an integer age as input.
- print "Eligible to vote" if the person is eligible to vote.

class IfElse {
```
public static void main(String[] args {
int age = 20;
if (age>18){
System.out.println("Eligible to vote.");
} }
```

Q.2. Take two integers A & B as input.
print the max of two.
Assume: x and y are not equal.
i/p:
x=5 y=7
output:
7 is greater.

actual code

```
if(x>y){
} SOP(x+"is greater");
else{
SOP(y+"is greater
}
```

my code

```
int A = 5
int B = 7
if ( B>A ) {
SOP("7 is greater");
```

12 37 codes आहे.

M	T	W	T	F	S	S

Page No.:
Date:
YOUVA

problem statement :-
- take two integers A and B as input.
- print the max of two.

input 1 :
x = 5 y = 7
output 1 :
7 is greater

Input 2 :
x = 5 y = 5
output 2 ;
Both are equal.

~~if (x > y) {~~

इथे आपल्याला if-else ladder use करावी लागेल. (शिडी)

```
if (x > y) {
    sop( x + " is greater");
} else if (y > x) {
    sop( y + " is greater");
} else {
    sop (" Both are equal");
```

problem statement :-
→ Given the temperature of a person in farenheit
- Print whether the person has high, normal, low temperature.

> 98.6 ⟹ high
98.0 <= and <= 98.6 ⟶ Normal
< 98.0 ⟶ low

class demo .


```
float ~~double~~ temp = ~~98~~ 92.6f

if( temp >98.~~6f~~){
    sop(" high")
}elseif( ~~temp~~ 98.0f<=temp <=98.6f){
    sop("Normal")
}else if ( ~~98~~temp < 98.0f){
    sop(" low")
}
```

✓

```
if (temp >98.6f){
    sop(" High")
}elseif( temp<98.0f ){
    sop(" low")
} else {
    sop (" Normal");
```

~~p~~roblem statement

take an Integer as input and print whether it is divisible by 4 or not.

Input : 5

output: Not divisible.

```
int x=5
if( x%4==0){
    sop(" ~~Is~~ divisible)
}else
{ sop(" Not divisible");
```

Date:

problem statement

Given an integer input, print whether its odd or even.

Input : 7
output : 7 is odd
Input : 4 ~~is even~~.
output : 4 is even.

```
int x = 7;

if (x%2 == 0){
    sop( x + " is even" );
} else{
    sop( x + " is odd" );
```

problem statement

- Given an integer as input.
- print fizz if it is divisible by 3.
- print buzz if it is divisible by 5.
- print fizz-buzz if it is divisible by both.
- if not then print "Not divisible by both"

```
int x = 5
if( x%3 == 0){
    ~~print~~ sop( "fizz" );
}else if( x%5 == 0){
    sop( "buzz" );
}else if( x%3 == 0 && x%5 == 0){
    sop( "fizz-buzz" );
}else{
    sop( "Not divisible by both" );
```

हा code बरोबर नाहि पण sequence matter करतो.

```
if( x%5==0 && x%3==0){
    sop("fizz-Buzz");
}else if (x%3==0){
    sop("fizz")
}elseif.(x%5==0){
    sop("Buzz");
}else{
    sop("Not divisible by both");
```

(8) Electricity Bill problem

- Given an an integer input A which represents units of electricity consumed at your house
- calculate and print the ~~amount~~ bill amount.

units <= 100 : price per unit is 1

unit > 100 : price per unit is 2

Input: 50

output: 500

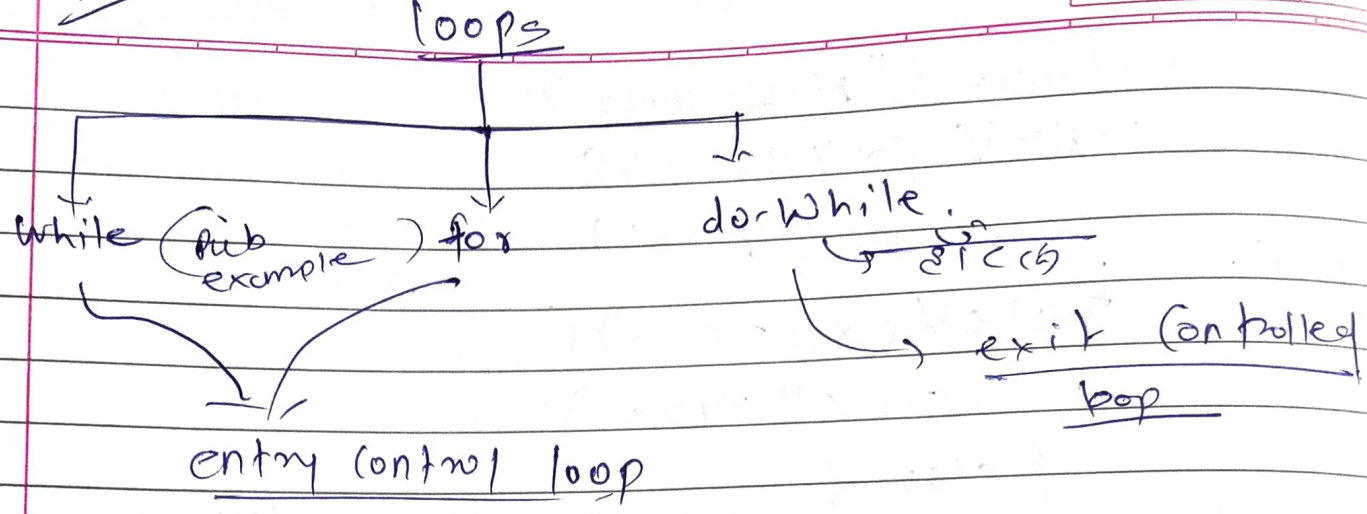
Input : 200

output : ? is 300

```
x = 50;
if( x <= 100 ){
    sop( x * 1);
else{
    sop(x*2);
```

```
x =
if (x <= 100){
    sop(units * 1);
}else{
    sop(100 * 1 + (unit-100) * 2);
}
```

1-5-23



1) While loop.

```
int i=1;
while(i<=5){
    sop("Hello");
    i++;
}
i++;
```

initialization
condition
body
increment or decrement

int ans=x++;
sop(ans)

output : infinite loop

Q.1 Print integers from 1 to 10 using while loop.

```
int i=1;
while(i<=10)
    sop(i)
    i++;
}
```

bad practice
same opn

```
int x = 1;
x = x++;
sop(x);
```

→ o/p => 2 ✗

→ o/p => 1 ✓

```
int x = 1;
int ans = x++;
sop(ans);
sop(x);
```

internally →

```
int post(int x){
    int temp = x;
    x = x+1;
    return temp;
}
```

temp [1]

x [1] 2

Q.2) Print 8 to 1

book
design pattern by dummies

GOF
Gang of four

```
int i = 8;          ——— ① सगळ्यात पहिल्यांदा होणार एकदाच.
while(i >= 1){      ②
    sop(i);         ——— ③
    i--;            ——— ④
}
```

code dry run करण खूप महत्वाचे है।

Dry Run

① i	② i >= 1	③ sop(i);	④ i--
8	true 8 >= 1	sop(8);	i=7
	true 7 >= 1	sop(7);	i=6
	true 6 >= 1	sop(6);	i=5
	true 5 >= 1	sop(5);	i=4
	true 4 >= 1	sop(4);	i=3
	true 3 >= 1	sop(3);	i=2
	true 2 >= 1	sop(2);	i=1
	true 1 >= 1	sop(1);	i=0
	✗ false 0 >= 1	~~sop( )~~	

Kadane's algorithm / in array

Page No.:
Date:

Q.3. Take an integer N as input.
Print odd integers from 1 to N using loop.

Input : 10
Output : 1, 3, 5, 7, 9

```
int N = 10
int i = 1
while(i <= N){
    sop(i);
    i = i+2;
}
```

1st approach

```
if(i%2 != 0){
    sop(i);
}
i++;
```

or (i%2 != = 1)

approach no. 2

Dry Run

N = 10

i	i<=N	sop(i);	i=i+2;
1	1<=10	sop(1)	i=3
	3<=10	sop(3)	i=5
	5<=10	sop(5)	i=7
	7<=10	sop(7)	i=9

Q.4 Take an Integer N as input.
Print multiples of 4 till N.

Input : 22
o/p : 4 8 12 16 20

N = 22

```
i = 4
While(i <= N){
    sop(i);
    i = i+4;
}
```

1st approch

or

```
if(N%i == 0)
sop(i);
i = i+4
```

2nd approach.

```
while(i <= N)
if(i%4 == 0){
    sop(i);
}
i++; or i=i+4
}
```

dry run

i=4	(i<=22)	sop(i)	i=i+4
	4<=22	sop(4)	8
	8<=22	sop(8)	12
	12<=22	sop(12)	16
	16<=22	sop(16)	20
	20<=22	sop(20)	24
false x loop मधून बाहेर	24<=22		

2-5-23

While loop :→

ex)

- Given an integer N.
- Print all its digit.

Input : 6531

Output :

1
3
5
6

पाहिजे तर %10

52 % 10 → 2

```
     5
10 ) 52
    -50
      2
```

```
N=6531
while(N!=0){
    sop(N%10);
    N=N/10;
}
```

Atlassian ~~कम्पनी~~. Adobe ~~कम्पनी~~ ~~की कम्पनी~~

10) 7 / 7

N = 753

```
while ( N != 0) {
    sop( N % 10);
    N = N/10;
}
```

N	N != 0	sop(N%10)
753	753 != 0	753%10 = 3
75	75 != 0	75%10 = 5
7	7 != 0	7%10 = 7
0	0 != 0	X

Q. Given an integer N.
Print sum of its digits.

Assume : N >= 0
I/P = 6531
O/P = 15.

my code

```
N = 6531
int sum = 0;
while ( N != 0 ) {
    sum = N % 10 + sum;
    N = N/10;
}
SOP( sum );
```

boolean (true or false)

dry run
~~multi = 1 N~~

for multiplication

I/P = 135

```
int multi = 1;
while (N != 0)
    multi = N%10 * multi
    N = N/10;
}
SOP( multi );
```

dry run

I/P = 135

multi = 1	N != 0	N%10 * multi;	N = N/10
1	135 != 0	135%10 * multi = 5 * 1 = 5	135/10 = 13
	13 != 0	13 %10 * 5 = 3 * 5 = 15	13/10 = 1
	1 != 0	1 %10 * 15 = 1 * 15 = 15	1/10 = 0
	0 != 0	X	

TLE → Time limit exceeded

GFG → code

- Take an integer N as input.
- print perfect squares till N.

Perfect Square : An integer whose square root is a integer.

25 → 5 : yes

81 → 9 : yes

1 → 1 : yes

10 → 3.13 : no.

i/p = 30

output : 1 4 9 16 25.

```
int N = 30
i = 1
while (i <= 30) {
```

① approach.

```
int N = 30;
int i = 1;
while (i*i <= N) {
    sop(i*i);
    i++;
}
```

dry run

~~le~~ $\frac{123}{10}$ 12.3

git → repository

Q. 4 Given an integer N.
- Reverse it
i/p = 6531
o/p = 1356

3-5-23

~~for loop~~ While loop.

H/w dryrun →

```
int N = 123
int rev = 0;
while(N != 0) {
int rem = N%10;
rev = rev*10 + rem.
N = N/10;
}
print (rev)
```

Q. Program 10; Write a program to check whether the number is palindrome number or not. (2332)
o/p = 2332 is a palindrome number

for & while are same.
while loop का compact version for loop है

H/W

dry Run

```
int N = 2332
int rev = 0;
int temp = N;

while( N != 0){
    int rem = N % 10;
    rev = rev x 10 + rem;
    N = N/10;
}
if ( temp == rev ){
    sop("Palindrome");
} else {
    sop( "Not palindrome");
}
```

while loop

```
initialisation ①
while(condition ②) {
    body ③
    updation ④
}
```

for loop.

```
for ( initialisation ①; condition ②; updation/incr/decr ④ ) {
    ③ body
}
```

Q.1 Take n as i/p. Print from 1 to N

I/p = 5
O/p = 1 2 3 4 5

```
int N = 5
for ( int i; i<=N; i++ ){
    sop(i);
}
```

dry run

i	i<=N	sop(i)	i++
	1<=5	1	2
	2<=5	2	3
	3<=5	3	4
	4<=5	4	5
	5<=5	5	6
	6<=5	X	

Date:

Q.2. Take N as input. print odd integers from 1 to N.

Input : 6
output : 1 3 5

```
int N = 6.
~~int i~~                    // i++.
for (int i=1; i<=N; i=i+2){

    sop(i);
    }
    // i++;
```

i	i<=N	sop(i)	i+2
1	1<=6	3	4

2 nd approach
or

$\frac{1}{3}$ ①
$\frac{3}{2}$

```
for (int i=1; i<=N; i++){

    if (i%2!=0){
        sop(i);
    }
}
```

$\frac{1}{2}$

OR i%2==1

Q.3. Take N as input. Print its factorial

i/p = 5
output = 120

i/p = 4
o/p = 24

Reading

M T W T F S S
Page No.: 24 YOUVA
Date: 5

2 | 120

```
int N;
int fact = 1
for (int i = 1; i <= N; i++){
    fact = fact * i;
}
SOP(fact)
```

dry run

fact	int i = 1	i <= N	fact = fact * i	i++
1	1	1 <= 5	1	2
2	1	2 <= 5	2	3
		3 <= 5	6	4
		4 <= 5	24	5
		5 <= 5	120	6
		6 <= 5	✗	

Q4. Take N as i/p. Print all its factors.

factors:

x is a factor of N if N % x == 0

i/p 1 : 6

o/p 1 : 1 2 3 6

i/p 2 : 24

o/p 2 : 1 2 3 4 6 8 12 24

Code

```
for (int i = 1; i <= N; i++){
    if (N % i == 0){
        SOP(i)
    }
}
```

i	i <= N	(N % i == 0)	SOP	i++
1	1 <= 6	6 % 1 == 0 true	1	2

Count print करने के लिए no. of factors,

```
int N = 6
int count = 0;
for(int i=1; i<=N; i++){
    if(N % i == 0){
        count++;
    }
}
Sop(count);
```

4-5-23

5-4-23

lect 18: for loop2

```
class forDemo{
    public static void main(String[] args){
        int x=10;
        for( int i=1; i<=10; i++); {

            sop(x+i);
        }
}}
```

हे i फक्त for ला दिसेल

या block ला नाही दिसणार

o/p: ~~11~~ ~~21~~

error : cannot find symbol.
sop(x+i);
symbol: variable i^
location: class forDemo.

→ array शिकायचं असेल तर for loop चा use करावा लागणं.

```
int x=10;
int i=1;
for( ; i<=10; i++);
    { sop(x+i); }
}
}
```

o/p = 21

1 to 1000 मध्ये perfect number किती आहेत ? आणि caution असतात

① Palindrome number
② prime number
③ perfect number
④ Armstrong number.

Q.5. Take N as i/p. count all its factor and print count.

Q.6. Take N as input. print whether its prime or not.

Quiz : if number (from 1 to N) is divisible by 1 and itself is called prim number.
A. true.
B. false. ✓

because 1 हा prime number nahi but वरच्या def^n नुसार हा prime number होता.

→ actual def^n : prime number हा असा number असतो ज्या नंबर चे factors exactly two असतात.

Code of prime number.

```
int N=5;
int Count=0;
for(int i=1; i<=N; i++){
    if (N%i==0){
        Count++;
    } }
if((count==2){
    sop("prime")
}else{
    sop("not prime");
```

Perfect number 4 → (1 + 2) . 4
3
≠

perfect (6) → 1+2+3 (6) ✓
(6)

Q. Take integer N as input print whether N is perfect numbers or not.

i/p : 4
o/p : not a perfect number.
i/p : 6
o/p : perfect number

2nd approach

```
int N=4
 int sum=0;
for (int i=1; i<N; i++){
      if (N%i==0){
         sum = sum+i;
      }
}
if ( sum==N){
        sop( "perfect number");
   }else{
        sop ("Not a perfect number");
```

Page No.:
Date:

~~यह हमको~~ student

Q.8. Take N as input print whether N is Armstrong number or not

i/p : 23
o/p :- Not an armstrong number

i/p : 153

o/p : Armstrong number

Code; (from myside) → imp.

```
int N=153
int temp =N; ~~temp~~
int sum =0;
for (int i=1; i<=N; i++){
        ~~int rem=N%10;~~
        sum =sum + rem×rem×rem
        N =N/10;
}
if(sum==temp){
        sop("ArmstrongNumber");
}else{
    sop( "Not armstrong number"));
```

java database value देत नाही

5-4-23 leet (19)

for loops

for loop मध्ये दोन condition करतोय नसतात.

Armstrong number रुकुण पुढच्या leet 19 चे Notes इथल्या वहीत आहेत.

forloop: 20
8-5-23

Microsoft CEO → satya nadella (tiny cook)

break & Continue theory
loop नं सांगायला.

(Gary kildar) — 105 [illegible]

```
class Demo {
    public static void main(String [] args){
        int x = 10;
        {
            int x = 20;
            System.out.println(x);
        }
        System.out.println(x);
```

error :- variable x is already defined in method main (String [])

Break & Continue :- ⇒

(real time example : [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible] [illegible].)

example :→

```
int N = 15;
int count = 0;
for( int i=1 ; i<=N ; i++) {
    if (N%i == 0) {
        count++;
    }
    if (count > 2)
        break;
}
if (count == 2)
    sop("prime");
}else {
    - - -
```

optimized code →

```
class BreakDemo {
    Public static void main(String[] args
        int N = 40;
        for(int i = 1; i<=N; i++) {
            if (i%3==0) {
                break;
            }
            System.out.println(i);
    } }
```

o/p => 1
2

Code:-

```
int N=40
for (int i=1; i<=N; i++){
    if( i%3==0 && i%5==0){
        break;
    }
    System.out.println(i);
}
```

* Continue: → skip current.

```
int N = 50;

for (int i = 1; i <= 50; i++){
    if((i%3==0 && i%5==0) || (i%4==0)){
        continue;
    }
    System.out.println(i);
```

if (i == 10)

9-5-23

Nested for loop

print ★ ★ ★ ★

```
int n=5;
for(int i=1; i<=n; i++){
    System.out.print("*");
}
```

o/p :- ★★★★★

```
for(int k=1; k<=3; k++){
    for(int i=1; i<=4; i++){
        soprint("*");
    }
    sopln();
}
```

Q. 7 7 7
7 7 7
7 7 7
7 7 7

```
for(int i=1; i<=4; i++){
    for(int j=1; j<=3; j++){
        soprint("7");
    }
    sop();
}
```

Q. 1 2 3
1 2 3
1 2 3

```
for(int i=1; i<=3; i++){
    for(int j=1; j<=3; j++){
        soprint(j);
    }
    sop();
}
```

Q. 1 1 1
2 2 2
3 3 3

```
for(int i=1; i<=3; i++){
    for(int j=1; j<=3; j++){
        soprint(i);
    }
    sop();
}
```

Q. 3 3 3
4 4 4
5 5 5

```
for(int i=1; i<=3; i++){
    for(int j=1; j<=3; j++){
        soprint(i+2);
    }
    sop();
}
```

10-5-23

Nested for loop 2 lect 21

patterns

1 2 3
4 5 6
7 8 9

A - 65
B - 66
Z - 90

a - 97
z - 122

```
int num=1
for(int i=1; i≤3; i++){
    for(int j=1; j≤3; j++){
        sop(num)
        num++;
    }
    sop( );
}
```

A B C
A B C
A B C

[A] B

```
for(int i=1; i<=3; i++){
    char ch= "A"
    for(int j=1; j<=3 j++){
        soprint(ch);
        ch++;
    }
    sop
}.
```

error: incompatible type
(ch=ch+1;)

string मध्ये एखादी value आहे किती शेजारच्यांना पण string बनवतो.

example:

```
int a=10;
int b=20;
sop("ans="+a+b);
```

o/p => ans = 1020

→ परंतु character ला बरोबर तसं नाही करत.

→ internally हा ch=ch+1; लागू होत नाही.

सगळ्यात कमी priority = ची आहे.

precedence

division, multipli, % → ह्या Same priority.

left to right जातो.

HW: string सोबत सगळ्यांना datatype देऊन बघा

A 1 B 2

A 1 B 2

A 1 B 2

```
for(int i=1; i<=3; i++){
    char ch='A'
    for(int j=1; j<=2; j++){
        sopnint(ch++' '+j+' ');
    }
    sop();
```

OR

```
for(int i=1; i<=3 i++){
    char ch='A'; int num=1
    for(int j=1; j<=4; j++){
        if(j%2==1){
            sop(ch++);
        }else{
            sop(num);
    }}
} sop();
```

A 2

A 1 B 2
C 3 D 4
E 5 F 6

```
int num=1;
char ch=A;
for(int i=1; i<=3; i++){
    for(int j=1; j<=4; j++){
        if(j%2==1){
            sopnint(ch++);
        }else{
            sopnint(num++);
        }
```

11-5-23

lect 03 Nested for loop 03.

N=4

~~for(int i=1;~~

```
* - - *
* - - *
* - - *
* - - *
```

N=3

```
* - *
* - *
* - *
```

```
for(int i=1; i<=N; i++){
    for(int j=1; j<=N; j++){
```

4%1==0
4%2==0
4%3==0 ×
4%4==0

~~if(N%j==0){ sop("*"); }else{ soprint("_"); }~~

```
if(j==1 || j==N){
soprint("*");
}else{
soprint("-");
}
}
sop();
}
```

Using %

```
if( j%N==1 || j%N==0){
  soprint("*");
}else{
  Soprint("—");
```

N=4

```
*
* *
* * *
* * * *
```

```
for(int i=1; i<=N; i++){
  for(int j=1; j<=i; j++){
     sop("*");
  }
  Sopritln( );
```

dry run

i=1 i<=N; j j<=i
1 1<=4; 1 1<=1 ✓
2 2<=1 ×
2 2<=4 1<=2 ✓
2<=2 ✓
3<=2 ×
3 3<=4 1<=3 ✓
2<=3 ✓
3<=3 ✓
4<=3 ×

4 4<=4 1<=4 ✓
2<=4 ✓
3<=4 ✓
4<=4 ✓
5<=4 ×

o/p:
```
*
* *
* * *
* * * *
```

Q. 1
2 3
4 5 6
7 8 9 10
11 12 13 14 15

```
int num=1;
for (int i=1; i<=N; i++){
  for(int j=1; j<=i; j++){
    sop(num+" ");
    num++;
  }
  sop();
}
```

i=1 i<=N; j j<=i num
1 1<=5; 1 1<=1 [1] 2
2<=1 ×
2 2<=5; 1 1<=2 ✓ [1]
2<=2 [2] [3] 4
3 3<=5; 1 1<=3 ✓ 5
2<=3 ✓ 6

```
1
1 2
1 2 3
1 2 3 4
```

```
for(int i=1; i<=N; i++){
    int num=1;
    for(int j=1; j<=i; j++){
        Sop(num+" ");
        num++;
    }
    Sop();
}
```

```
10
9 8
7 6 5
4 3 2 1
```

```
int num=10
for(int i=1; i<=N; i++){
    for(int j=1; j<=i; j++){
        Sop(num+" ");
        num--;
    }
    Sop();
}
```

i<=N j j<=i
1<=

```
* * * *
* * *
* *
*
```

```
for(int i=1; i<=N; i++){
    for(int j=1; j<=N-i+1; j++)
        Sop("* ");
    }
    Sop();
}
```

N-i+1 = 4
N-1+1 = 4
4-1+1 = 4
N-2+1
4-2+1 = 3
2+1=3

2nd approach

```
for(int i=1; i<=N; i++){
    for(int j=N; j>=i; j--){
        Sop("* ");
```

j=N
j=4
4>=1 ✓
3>=1 ✓
2>=1 ✓
1>=1 ✓
0>=1 ✗

```
*
* * *
* * * * *
* * * * * * *
```

```
A B C D
A B C
A B
A
```

```
char ch='A';
for(int i=1; i<=N; i++){
    for(int j=N; j<=N-i+1; j++)
        Sop(ch+" ");
        ch++;
    }
    Sop();
```

13-5-23
seen

Page No.:
Date:

switch statements

↳ if else ladder सारखं च असतो.

```
int x = 2;
switch( x ) {

    case 1:
        sop("one");
    case 2:
        sop("two");

    case 3:
        sop("three");
    default:
        sop("other");
```

```
int x = 3;
if(x == 1){
    sop("one");
}else if(x == 2){
    sop("two");
} else if(x == 3){
    sop("three");
}else{
    sop("other");
```

menu card सारख्या गोष्टीसाठी लिहला जातो.

```
int x = 3;
switch (x) {
    case 1;
        System.out.println("one");
    case 2:
        System.out.println("two");
    case 3:
        ~~Syst~~ sop ("three);
    case 4:
        ~~sop("three")~~; sop(" four"
    Case 5:
        sop( "five");
    default:
        sop(" No match");
};
O/P
sop("After switch");
```

o/p :→ three
four
five
~~sop~~ After switch

| three
four
five
No match
Afterswitch

→ Answer fkt 3 यायला पाहिजे.
→ त्यासाठी break Picture मध्ये येतो
→ break loop ~~मध्ये~~ आणि switch चे block ~~br~~ तोडून टाकतो. आणि त्याच्या बाहेर पाठव

```
int x = 3;
switch(x) {
    case 1;
        sop("one");
        break;
    case2;
        sop(" two");
        break;
    case3:
        sop(" three");
        break;
    case 4:
        sop(" four");
        break;
    case 5: sop(" five");
        break;
    default:
        sop(" No match");
        break;
}
sop( "After switch");
```

least priority for default ← default

कुठल्याही लिहा बाहेर कुठेपण.

o/p : three
After switch.

```
class SwitchDemo{

    Public static void main(String[] args){
        int x=5;
        switch (x){

            case 1;
                sop("1");
                break;
            case2:
                sop("2");
                break;
            case5:
                sop("first-5");
                break;
            case5:
                sop("second-5");
                break;
            case2:
                sop("second-2");
                break;
            default:
                System sop("No match");
        }
        sop("After switch");
    }
```

✓

✗

o/p:- first-5
After switch.

error: duplicate case label
case 5:

error: duplicate case label
case 2;

2 errors

```
class ~~static~~ SwitchDemo {
    public static void main(String[] args){
        int ch = 65;
        switch(ch) {
            case 'A' :
                ~~System.o~~
                sop("char-A");
                break;
            case 65 :
                System.out.println("Num-65");
                break;
            case 'B' :
                sop("char-B");
                break;
            case 66 :
                sop("Num-66");
                break;
            default :
                ~~sop("Num-66");~~
                sop("Invalid")
                break;
        }
```

✓

o/p :⇒ ✗ Char-A

error : duplicate case label
case 65 :
error : duplicate case label
case 66 :
2 errors

~~आतांनी सार~~ नेवट switch statement
त्याच्या बाहेर switch आहे.
एवढंच

M T W T F
Page No.
Date
YOUVA

```
public static void main(String [] args){
    int x = 3;
    int a = 1
    int b = 2;
    switch(x){

        case a:
                sop("1");
            break;
        case b:
                sop("2");
            break;
        case a+b:
                sop("3");
            break;
        case a+a+b:
                sop("4");

        case a+b+b:
                sop("5");
            break;
        default:
                sop("Invalid");
            break;
    }
```

variable नको इथे किंवा constant value पाहिजे.

o/p:- error:

(error)
constant expression required
case b:
     ^
case a:
     ^

C मध्ये Char & integer fkt चालायचं

1.6 पर्यंत हे चालत नव्हतं.

ही code version वरती depend आहे.

```
String str = "mon";
switch(str) {
  case "mon":
        sop("Monday");
        break;
  case "tue":
        sop("tuesday");
        break;
  default:
        sop("It's sundy: ))))))");
        break;
  }
}
```

o/p: monday.

How to check version

sudo update-alternatives --config javac

16-5-23

25 : Nested switch statements.

जसेतं ~~साठ~~ Java साठी म्हणजे backend developer झालाच पाय नाही - java ला connect आहे

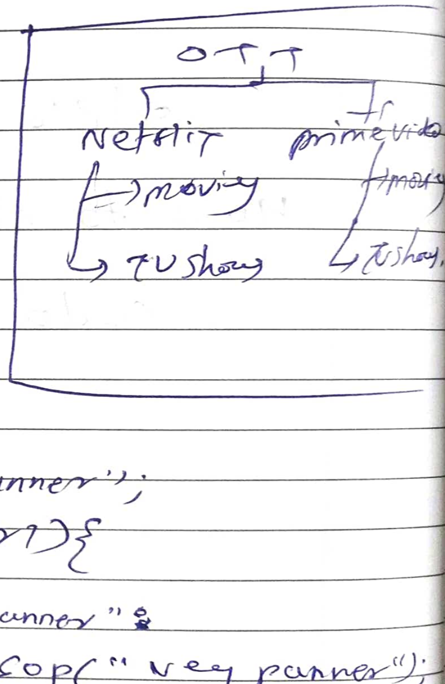
Java - 70% + spring boot / Nodejs - 30%

Backend developer

कोणताही OTT platform बघा सगळे: Java आहे त्यात.

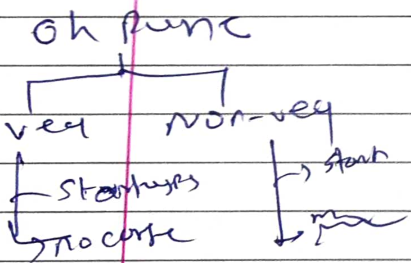
इथं limit पर्यंत Nested switch वापरतो.

example:



oh pune

```
sop(" oh pune");
string str = "veg"
switch (str) {
    case "veg":
        str1 = "panner";
        switch(str1){
        ① case "panner":
            sop(" veg panner");
        ② case "Dudh":
        }
    break;
    case "Nonveg":
        str1 = "matan";
        switch(str1){
            case "Anda":
                sop("Nonveg Anda");
            case "matan":
                sop("Nonveg matan");
```



H/w real time example write करो।

Sir call to jeevan

I/O – O/Scanner

Input-output In Java.

company
- study on layoffs
- GithuB company
- push/pull in github.

GithuB ← Take over by microsoft.
- microsoft la autopilot आणायचा होता. copilot
  - beta version

same → as chatgpt

① Sir talk about Data science.
① Python input कसा होता.

② OpenAI → owner

Input/output

C	C++	Java	Python (return type string)
I/p - scanf()	cin	Scanner class, Buffer reader	input ( )
O/p : printf()	cout	s.o.print(), println(), s.o.printf()	print ( )
Function	हे fn nahit object आहेत.	fn आहेत	fn.

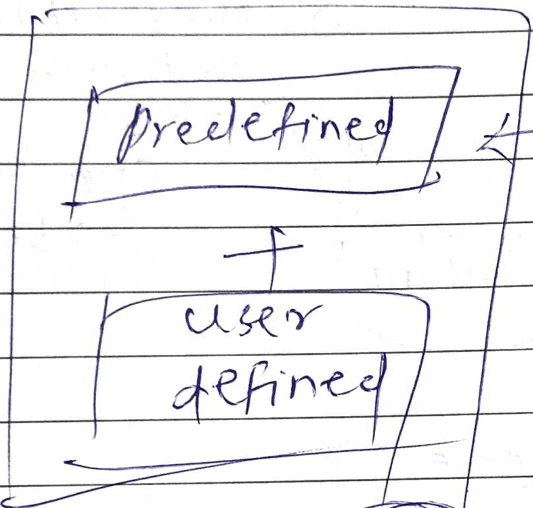
→ हे विसरू नका jvm एक process आहे

→ आणि आपला code कधीच process म्हणून जात नाही कारण एका process-च्या ~~potat~~ पोटात दुसरी process नाही जात

~~किंवा~~

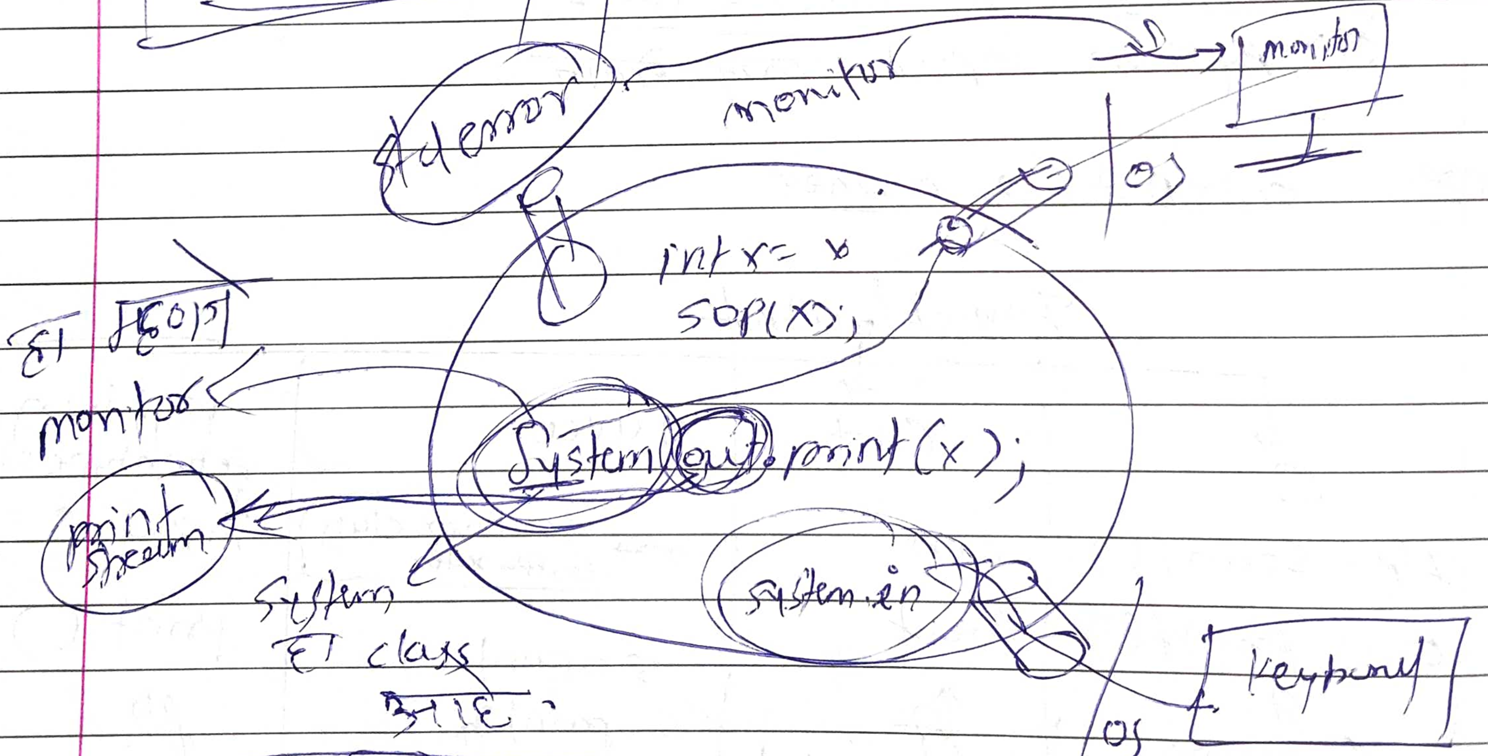
→ एका process-च्या पोटात thread असतो

→ आपला code तो एक jvm चा part असतो



हे जर java ने केले नसते ना तर आपला प्रत्येक code.

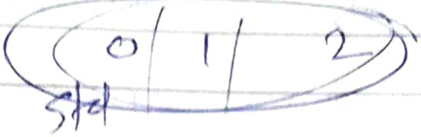
→ 1250 line च्या वरती झाला असता.

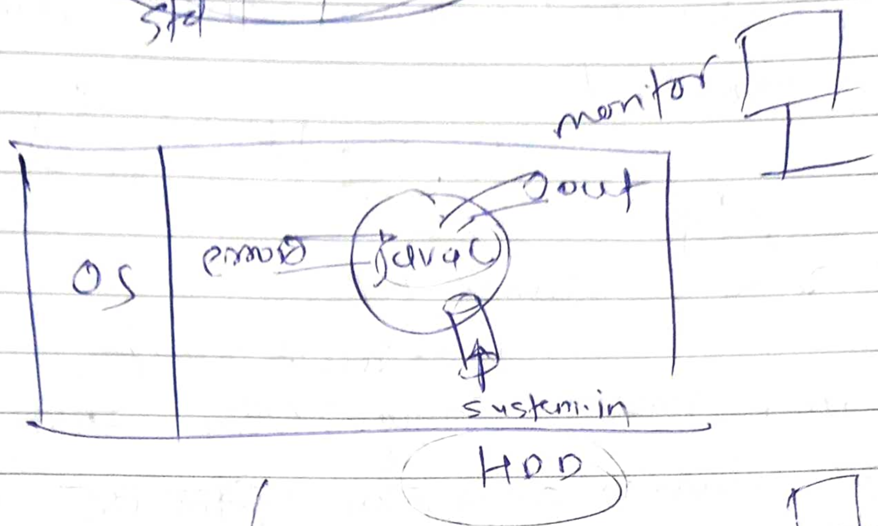


भारी गोष्ट

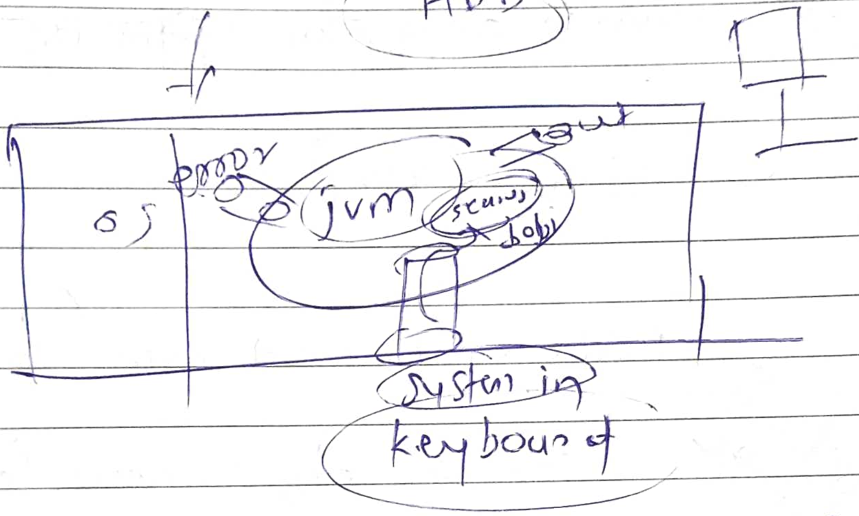
System. error. println पण fix असतं.

कुठलीपण process आणि तिला 3 pipe असतात
दिली असतात



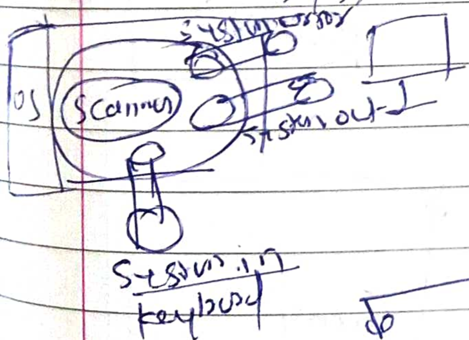


example format error
if we open .class file through vim



scanner obj = new Scanner(System.in);

System class खूप important in Java



I/P Java

- Scanner
  - System.in
- Buffered reader
  - Input streamReader
    - System.in

→ Scanner हा util package मध्ये आहे

→ Utility हे Package मध्ये add करावं.

multithreading is important.

Backend developers साठी multithreading खूप important.

19-5-23 (seen) → 17-5-23 uploaded

27 : Scanner i/p

(In default format)

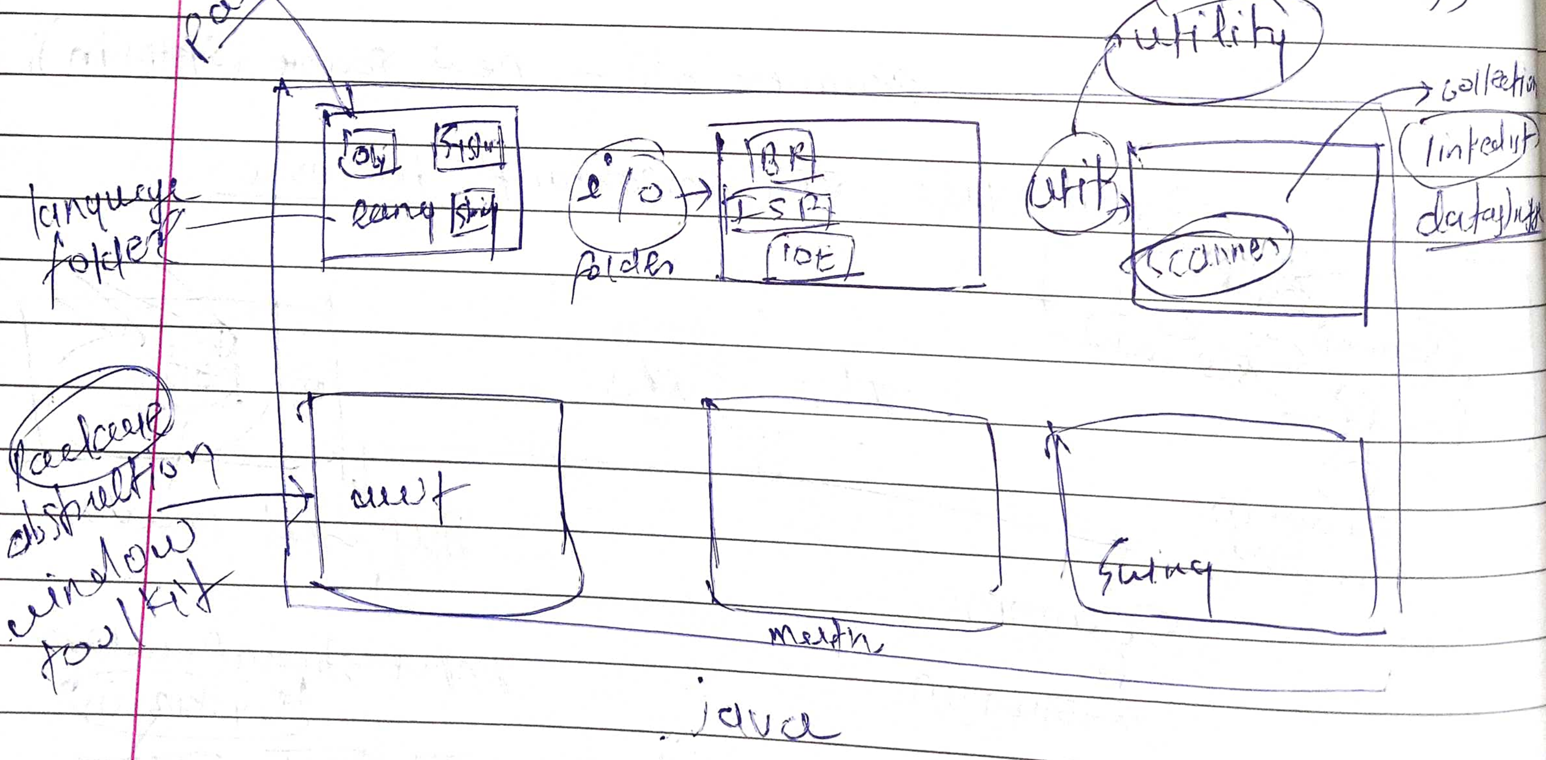
⇒ i/p → कायम String स्वरूप घेतो

जर String i/p String नसेल तर Next Int() function.

class ScannerDemo{

Public static void main(String[] args){

Scanner obj = new Scanner(System.in);



object class हा parent class आहे तो दिसत नाही.

is-a

Inheritance हा by default आहे

Scanner हा नंतर आला आहे 1.5 मध्ये.

System.in हा static variable आहे

keyboard connection

```
import java.util.Scanner;
class DreamCompany {
    p s v main(String[] args) {
        Scanner obj = new Scanner(System.in);
        System.out.println("Enter your Dream Company:");
        String name = obj.Next();
        System.out.println("Enter your Salary:");
        Float package = obj.NextFloat();
        Sop(name);
        Sop(package)
```

// Small n

हा class

// Camel Casing function Name

o/p → Enter your Dream Company:

Amazon

Enter your Sallery:

12.5 LPA

Javap java.io.InputStreamReader

For seeing methods in given class

M T W T F S S
Page No.:
Date:
YOUVA

BufferedReader → io package

↳ BufferedReader keyboard शी connection थेतच नाही

↳ त्यासाठी InputStreamReader Picture मध्ये येतो. आणि keyboard शी connection बनवतो

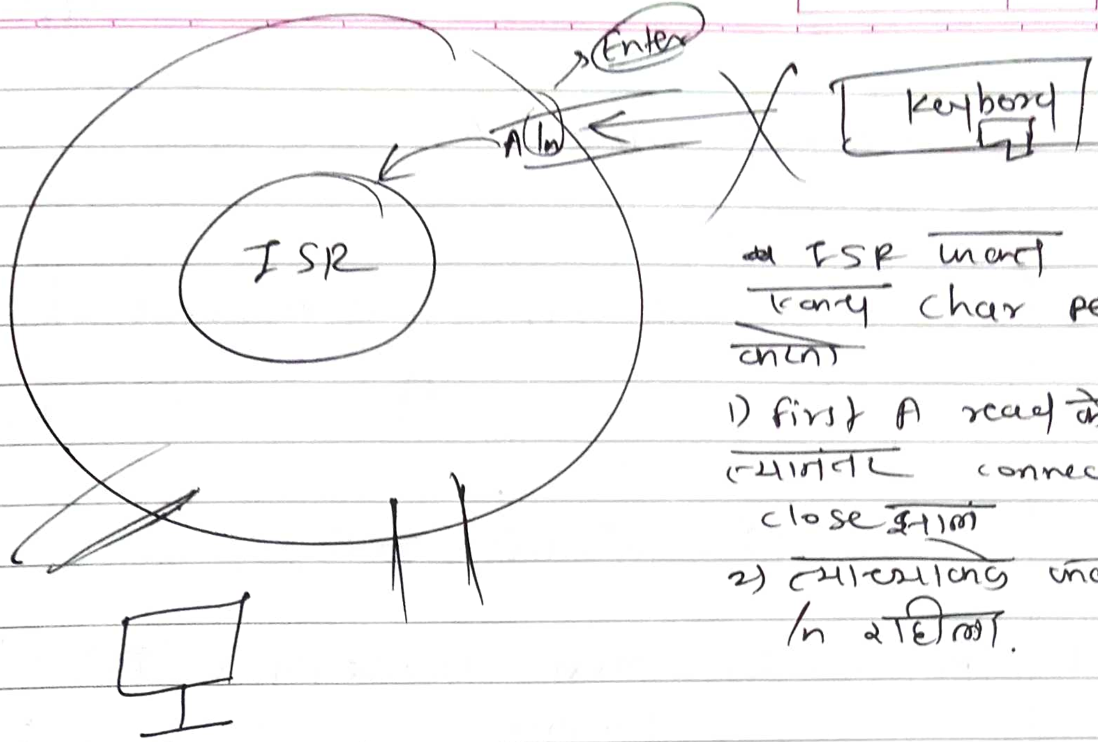
पण ISR फक्त एक character read करत शकतो.

ISR class मध्ये Read method ची return type int आहे.

```
import java.io.*;
class InputDemo{
    p s v main(String[] args) throws IOException
    {
InputStreamReader isr = new InputStreamReader(System.in);
System.out.println("Enter char");
int ch = isr.read();
// char ch = isr.read(); error incompatible
// char ch = (char)isr.read();
sop(ch);
isr.close()
sop("Enter char");
int ch = isr.read();
    }
```

Error:→
Exception in thread "main" java.io.IOException: Stream closed

IOException check the connection of keyboard



ISR मध्ये एकच char read करतो

1) first A read केला त्यानंतर connection close झाला

2) त्यानंतर मध्ये /n राहिला.

multiple character असतील तर Buffer Reader picture मध्ये येतो.

```
Public static void main(String[] args) throws IOException{

InputStreamReader isr = new InputStreamReader(System.in);

BufferedReader br = new BufferedReader(isr);

    sop("Enter Name");
    String name = br.readLine();      --> return as string
    sop(name);
    sop("Enter Age");
    int age = Integer.parseInt(br.readLine());
                  wrapper class   Typecasting
    sop(age);
}
```

Scanner class मध्ये आत असते parsing केला आहे. NextInt() f^n मध्ये

lect 28 (19-5-23) seen (18-5-23) uploaded

Page No.: Date: YOUVA

```
import java.io.*;
class Demo{
    public static void main(String[] args)
        throws Exception{

InputStreamReader isr =

BufferedReader br= new BufferedReader(new InputStreamReader(System.in
    sop("Enter Batsman name);
    String name= br.readline();
    sop(name);
    sop("Enter boller name);
    String name2= br.readline();
    sop(name2);
}
```

→ concept

For जर class आणि primitive data types conversion

अशा वेळी then error: incompatible type

error:- String cannot be converted into int

int jerNo:- br.readLine();

→ आणि float to int (or) int float

असेल तर

error incompatible types

possible lossy conversion

wrapper class Integer → parseInt method वापरून string to integer convert करतात.

```
import java.io.*;
class playerInfo {
    public static void main(String [] args) throws
                IOException{

    BufferedReader br = new BufferedReader(new InputStream-
        Reader(System.in));
        sop("Enter player name");
        String name1 = br.readLine();
        sop("Enter JerNo");
        int JerNo = Integer.parseInt(br.readLine());
        sop("Enter Avg");
        float avg = Float.parseFloat(br.readLine());

        sop(name1);
        sop(jerNo);
        sop(avg);
```

import java.io.*;

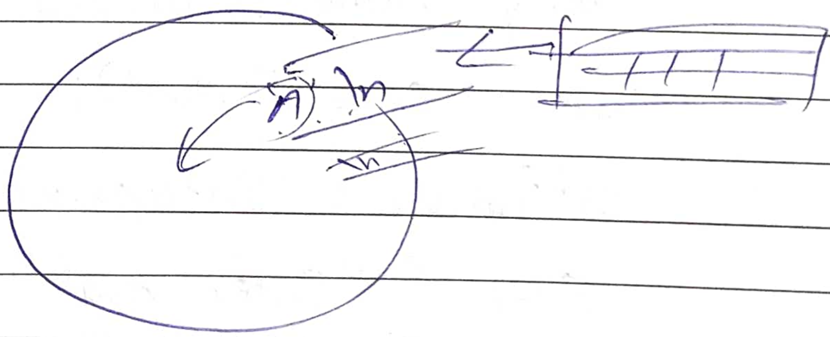
~~p s v main( ) throws IOException {~~

```
BufferReader br = new BufferedReader(new
    InputStreamReader(System.in));

    sop("Enter building name");
    String Name = br.readLine();

    sop(" Enter wing of building");
    char ch = (char) br.read();
    br.skip(1);          ← solution

    sop(" Enter Flat no.");
    int flatNo = Integer.parseInt(
                        br.readLine());

    sop(name);
    sop(ch);
    sop(flatNO);
}
```

o/p :→ error : number format ~~error~~ Exception

character जो शोषक घातला आहे.

कारण पाईप मध्ये \n तसेच राहील.

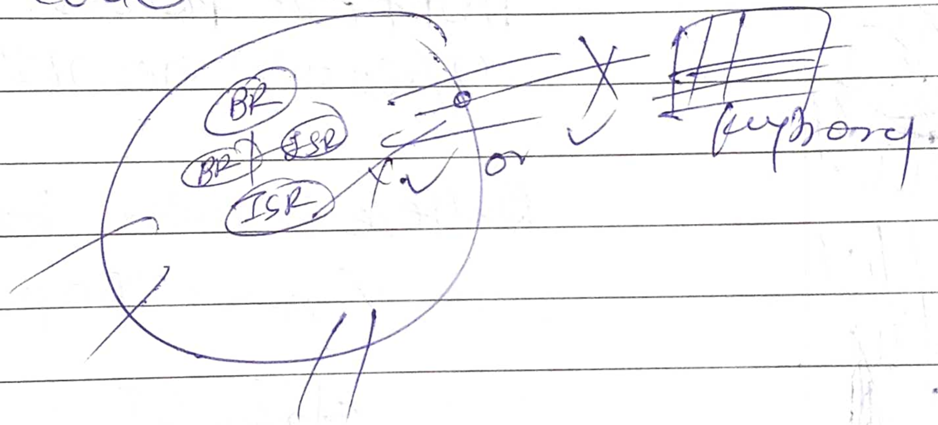


readline काय करतो \n पर्यंत असलेले वाचतो (Read)

read ची एक char वाचायची capacity आहे.

```
Buffered Reader br1 = new BR( new ISR(System.in));
———"———         br2 = new BR( ——"—— );
String str = br1.readLine();
       br1.close();
String str2 = br2.readLine();
```

Try this code



29

19-5-23 Unlocked → 20-5-23 seen

M T W T F S S
Page No.:
Date:
YOUVA

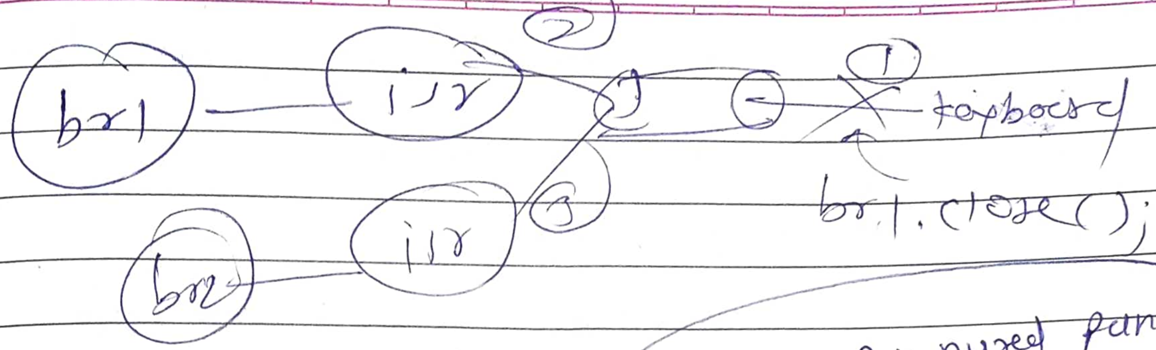
BufferedReader

```
import java.io.*;

class IODemo{
    public static void main(String[] args)
                throws IOException{

BufferedReader br1 = new BufferedReader(new InputStreamReader(System.in));

BufferedReader br2 = new BufferedReader
        (new InputStreamReader(System.in));

String str1 = br1.readLine();
System.out.println("String1 = "+str1);

String str2 = br2.readLine();
SOP("String2 = "+str2);
    }
}
```

error: unreported exception IOException;
must be caught or declared to be thrown.

```
String str1 = br1.readLine();
              ^
```

jvm have one ~~मैनेजर~~ that's name is default exception handler



book :- Big nerd ranch
Android शिकणा अनुवाद.

p s v main(String [] args) throws IOException

InputStreamReader isr = new InputStreamReader (System.in)

BufferedReader br1 = new BufferedReader(new InputStreamReader(System.in));

String str1 = br1.readLine();
System.out.println("String 1 = "+str1);

char ch = (char) isr.read();
sop("char = "+ch);

~~runtime~~ error :
runtime

Exception in thread "main" java.io Exception: stream closed
+ stack trace

Page No.:
Date:

Question.

sop ("Enter matchinfo, mom player & runs");

i/p KKRGT Rinku 48

o/p : matchinfo : KKRGT

mom player : Rinku Singh,

Runs : 48.

Solution:

```
import java.io.*;
class IODemo{
public static void main(String [] args) throws
                    IOException{

BufferedReader br = new BufferedReader(new InputStreamReader
                                            (System.in)));

    sop("Enter match info, mom player & runs");

    ~~KKR~~
        String str = br.readLine();
```

* Icon वाली (class) picture ~~है~~ (String Tokenizer.)

Tokenization


```
* import java.io.*;
  import java.util.*;
class StringDemo{
        public static void main(String [] args) throws IO Exception {

            BufferedReader br = new BufferedReader(
                        new InputReader(System.in));

            Sop( "Enter Society name, wing,
                   Flat no");

            String info = br.readLine();
            Sop( info);

          StringTokenizer obj = new StringTokenizer
                        (info, ",");

          String token1 = obj.nextToken();
           String token2 = obj.nextToken();
            String token3 = obj.nextToken();

          sop( "Society=" + token1);
          Sop( " wing =" + token2);
          sop( " Flat =" + token3);
     }
}
```

readLine example
string
char
int
float

Hata
SZIIZEU)
usrancon.
use
charAt(0);

uploaded 22-5-23

seen at 25/05/23

lect 30

c चा program चे process करणे आहे.

~~हे compile time ला समजेल आहे.~~

हा run time ला बनतो

Scanner sc = new Scanner(System.in)

~~left side~~ ~~right side~~

compile time ला

~~जशी buffer reader मध्ये readline नावाची method आहे तशी scanner मध्ये nextLine आहे.~~

hasMoreTokens()

~~hasMoreElements();~~

```
Scanner sc = new Scanner(System.in)
String str= sc.nextLine();
StringTokenizer st = new StringTokenizer(str, "_")
    Sop(st.countTokens());
    While (st.hasMoreTokens()){
        Sop(st.nextToken());
    }
}
```

String

Delimeter

functions

Methods in Java

There are two types of method

method
- ↓ static
- ↓ non-static / Instance

```
class MethodDemo {
    public static void main(String[] args) {
        fun();
        gun(); → error
    }
    static void fun() {
        Sop("In fun method");
    }
    void gun() {
        Sop("In gun method");
```

↳ there is no directly call for nonstatic method

error → non-static method gun() cannot be referenced from a static context
gun();
^
1 error

23/5/23

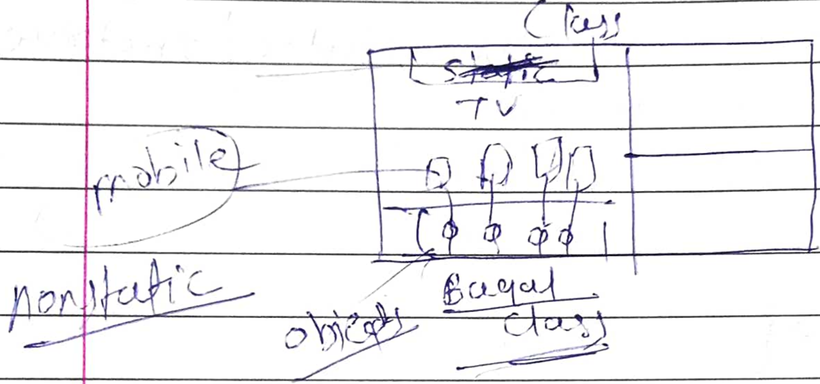
lect 31 seen 26-5-23

Method (2)

for redundency method are used

class से variable अक्सेस करना तो static

object से अक्सेस करना non static



object का bind अक्सेस करना nonstatic region.

```
class Demo{
        int x=10;
        static y=20;                  void f
    ~~public static void main(String[] args)~~
    void fun(){
            Sop("In fun method");
    }
    static void gun(){
            Sop("In gun method)
```

instance of object [illegible] nonstatic [illegible] → [illegible] object [illegible] initialize [illegible]

```
public static void main(String [] args){
    Demo obj = new Demo()
    sop(obj.x)

    sop(obj.y)
```

```
class Demo{
    int x = 10;
    static int y = 20;
    void fun(){
        sop(x);
        sop(y);
    }
    public static void main(String[] args){
        Demo obj = new Demo()
        obj.fun();
    }
```

O/P: 10
20

passing values to a functions

```
import java.util.*;

class Add{
    static void add(int a, int b){
        int ans = a + b;
        Sop(ans);
    }

public static void main(String[] args){

    Scanner sc = new Scanner(System.in);
    Sop("Enter int values");
    int a = sc.nextInt();
    int b = sc.nextInt();
    add(a,b);

    sub(a,b);
    mult(a,b);
    div(a,b);

}
```

```
static void sub(int a, int b){
    int sub = a-b;
    sop(sub);
}
static void mult(int a, int b){
    int mult = a*b;
    sop(mult);
}
Static void div(int a, int b){
    int div = a/b;
    sop(div);
}
```

b. putting above code using function

lect 32 year 26-5-23 24/5/23 upload

```
class Demo{
    void fun(int x){
        sop(x);
    }
    public static void main(String[] args){
        sop("in main");
        Demo obj = new();
        obj.fun()
        sop("End main")
    }
```

error: method fun in class Demo cannot be applied to given types;
obj.fun();
required: int
found: no arguments
reason: actual and formal argument lists differ in length

1 error