

Core Java

Q) Java class will have main?

 throws main method

 not found in class

 Q) JVM searches for main method, compiler don't look for it

 Q) no compile time error

 Q) follow standard starting point

 PSV main(String[] args){

}

 Q) can we write multiple Java class and with multiple main method?

 Yes, it will generate A.class, B.class

 java A, java B to run

 Q) Should public class name

 match with file name?

 Yes and there should be only

 one public in one file

.java

 Q) What does jar file consist of?

 i) consists of bundled java classes

 java.lang
 util

 How much byte does boolean

 occupy.

 Q) It occupies i.e 1 or 4 byte as

 per JVM specification

 internally it stores 0 or 1 hence

 0 or 1 may be int, short,

 char \rightarrow 2 Byte

 Q) How is long - float \rightarrow why possible auto conversion

 long \rightarrow 8 byte long to float conversion

 float \rightarrow 4 byte

 the result in loss of precision i.e result may loose 00m LSB digits bits in such case number is correctly rounded

 byte b₃ = b₁ + b₂ (where b₁ & b₂ are byte)
is error? TestConversions.java

 b₁ & b₂ \rightarrow get converted to int and int to byte is not auto conversion

 float d₂ = 110; no error auto conversion

 float d₃ = 10.34; error \rightarrow

 b₁ + b₂ \rightarrow no error \Rightarrow adds implicit

 type cast

b₁ $\overset{\text{byte}}{=}$ (b₁ + b₂)

What is scanner \rightarrow readme_day

Scanner sc = new Scanner(System.in);

default delimiter = space

nextline \rightarrow delimiter is nextline, \n

What is class and what is real life example
class is blue print to create object
which tells what all data members
and functions will be in object.
 \Rightarrow it binds data members to functions
together

\rightarrow building house

 class \rightarrow consult architect & tell our specification
 Blue print \rightarrow house (class)
 (2D, 3D) \rightarrow approval
 not a house \rightarrow (Object) =

 Object \rightarrow represents real life entities
 \rightarrow house (neighbours)

 what does object has

 ① state ② behavior ③ identity

What is use of this?

→ current object's first line

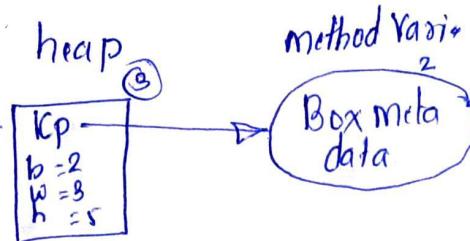
→ to have or call constructor from other constructor

→ `int a`

`this.a=a;` → to differentiate object variable & argument
Unhide Inst var from : local var

`Box b1;`
`b1 = new Box(2, 3, 5)`

stack
`b1`



Can we request JVM for garbage collection?
Yes we can hint JVM by System.gc() method

(↳ 100% guarantee is not final if gc will work)

④ gc calls finalize method before destroying Object (↑ it is method of Object class)

When will method area deallocated?

When JVM terminates all free memory deallocated

`Box b1 = new Box();`

`Box br = new Box(4, 5, 6);`

How many Garbage collected

day 3 P.

What is package? and what is need of creating it?

④ collection of functionally similar classes & interfaces.

need → To group functionally similar classes together

① avoid namespace collision

② flow control over access specifier

→ packages → mapped to folder

④ package name must match with folder name

src

↳ P1

from src

`javac -d ..\bin p1\A.java`

is it compulsory to maintain package structure under src?

it is not mandatory to create java sources (.java) under package named folder. But its mandatory to store packaged compile classes (.class) under package name folder

e.g. package (P1)

↳ class A { }

day 3.1

↳ src
↳ com

↳ sunbeam
↳ cori

compile from src

↳ sunbeam
↳ cori
↳ box.java
Package com.
sunb. cori

com\sunbeam\cori\box.java

→ testor

↳ test.java
Package com.testor

Compile only testor

import package.class

④ Compiler automatically creates folder structure

To run → use fully 'Q' class name
com
java. com. subram. core. tester. java
↳ compulsory.

when package is used compulsory
Fully qualified name

⑧ also fully Q class name must
match with folder structure

⑨ run from bin only

bin → run from here
↳ com
↳ tester

why we don't write .class
java test.class?

because when we run with
package e.g. com. subram. sum. class
it will search for folder structure.
→ package separator

How to run code from anywhere?
set classpath ^{called as} environment variable

and IDE sets this variable auto
matically.

As what should be value of
class path?

must be set to top of package
hierarchy

bin →
↳ package structure

java -cp "path till bin" com. tester. sum
" (e1.../bin"

To set permanently → set environment
variable

classpath
name = Value; Value2;

Explain everything in java is
pass by reference?

emp e1, emp e2 → same

swap(e1, e2) → no changes

swap(Emp e1, Emp e2) ↳ here copied

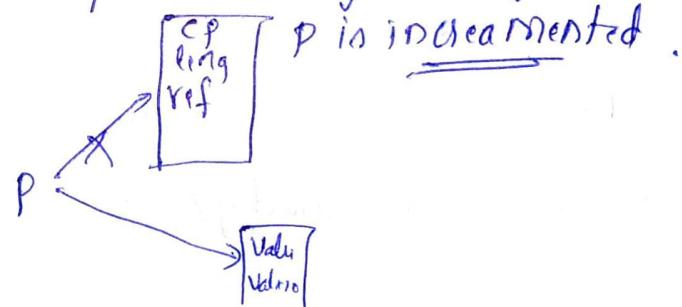
e1, e2 → are same.

What is accessibility of protected
They are accessible in package
whether class is sub class or not

What is meaning of for each works on
copy?

```
Point2D[] points = new Point2D[3];  
int value = 10;  
for (Point2D p : points) {  
    p = new Point2D(value, value + 10);  
    value++;  
}
```

here no change b/c address gets copied
in p and changes are ignored in when
p is incremented.



What you know about static?

→ loaded at download

→ only one copy is maintained

→ supports overloading

→ static initialiser for initing objects

→ In method area.

→ static cannot access non static

b/c they are created @ runtime
but static are available at class
loading

What is difference b/w import & static import?

Import → used to import class from particular package

Import static → is used to import static members of a class
java.lang.math.PI/*.

What is static initialiser block & non static initialiser block

Static { multiple blocks allowed }

① gets called only once @ classloading by JVM

② to init all static members

③ to add functionality that is called only once

④ to load hibernate/spring framework heavy

{
→ called per instantiation
→ before constructor
}

Inheritance → readme → inheritance

Constructor invocation hierarchy? ↗
parent → child → child →

What will happen in following case? if parameterized const not written ↗

Person → fName, lName

Person → gradYear, course
Super (fn, ln) ↗
fees, marks. ↗ parametrized constructor

① default const added by Compiler

②

Upcasting → readme

What is Diamond Problem

class A {

show()

class B {

show()

}

class C {

show()

}

class D {

show()

}

What does toString of object class return?

For class name @ java.lang.toString

What is runtime polymorphism
decision of which method to be called on which object is taken at runtime by JVM called as runtime polymorphism

Object take multiple forms

Method overloading →
same method name different argument, or diff no of argument
return type ignored by compiler
e.g. print(char/boolean/float)

→ Static methods can be overloaded

Arrays.toString(int[]/char[]/byte[])

8:11

- If no matching constructor is not found then compiler searches for nearest type method
 - byte → short → int → long → float → D
- Slightly Wider
- * long to float auto promotion
- Digram → ~~day-6~~ overloading vs overloading vs overriding
- Why static methods cannot be overridden?
 - readme → Why static methods
- They are related to class and not object hence cannot be overridden
- ↳ down casting is required for Java compiler only
- When we require downcasting explicitly?
 - When super class object pointing to sub class & trying to call subclass specific functionality
- Best example of abstraction?
 - ATM
 - java is fully encapsulated
 - we cannot write outside the class
- What is polymorphism?
 - multiple changing forms
- Which methods can be overridden?
 - Virtual methods, ~~No virtual keyword~~
 - In java all methods are implicitly virtual except static, private & final.

- Which methods bound early?
 - ~~private, static, final, overloaded~~ methods.
- Which are late bound?
 - methods that are not P, st, f, o
- privat + abstract
- final + abstract
- Both not allowed
- Which are standard final methods?
 - wait, notify, notifyAll
 - standard final class?
 - Stringbuffer, StringBuilder,
- What is interface?
 - blueprint for class implementation
 - ↳ to achieve 100% abstractn
 - ↳ shows is-a relationship
 - ↳ to achieve loose coupling.
- Who differentiates checked and unchecked?
 - Java compiler.
- Checked → compulsory implementn
- unchecked → handled by JVM
- Why try{} catch(Exceptn e){}
is catchall block?
 - ↳ inheritance hierarchy Exceptn
- Exceptn can hold runtime compil.
all Exceptn due to upcasting
- throws vs throw
 - readme 8th day help
- When is throws mandatory?
 - unhandled checked Exe
 - no try catch

difference b/w finally & finalize

final → const

finalize → object method

finally → block of try (always executes)

→ always executed after
return also

④ finally does not handle Exception

try with resources can exist without catch?

yes try(?) {
scanna-

④ Scanner sc = new(); *

④ → FileReader fr = new File("...")
→ closed fr.close()

Debug → day 8 lab

Differentiate StringBuilder vs StringBuffer

④ Difference between equality vs content equality?

s1 = new String("test");

s2 = new String("test");

s1 == s2 → false → reference

s1.equals(s2) → true → content

HashCode of String?

HashCode generated as per content
if string content equal than same
HashCode

StringBuild equals class?

does not override equals.

s3 = "He" + "llo"

s4 = "He".concat("llo")

s3 → points to literal string in
the pool

s4 → points to non-literal

"He", "llo", "Hello" added to
pool

④ what are enums? and what
are their uses.

④ can we extend class from enum?
No they are final & decompiled
public final class Category
extends Enum {

{ bad file

common class → java.lang.Enum

④ Compiler synthesizes huge
code behind

④ constructor is protected

protected Enum(const, ordinal pos)

④ it has private constructor

④ Why Enum over interface.

In interfaces we cannot loop over
all constants whereas looping
is possible in enum

④ category.getValues() method

④ P.S. Category ValueOf(String),
E = E + Interface

Q) enum consist of self types

final Category Petrol

final Category diesel

Petrol = new Category("Petrol", 0);

which methods in enum can be overridden?

To String, comparisons, equals can not be overridden

Implemented if → Comparable

→ we can add our own compare
noodles(60, breakfast)

Why enums?

① helps define constants

② add type safety to constant

③ situation over constant is possible
that is not possible with interface

④ can be treated as class as well

⑤ we can add state as well with
constants.

When outer class is instantiated

does the inner class also gets instantiated

→ No we need to do explicitly

→ inner can access outer class members

→ but outer cannot access inner members

Outer.deliveryAddress ad = new VehicleInfo()

new delivery()

inner class → in day 11.3

Outer.java

Outer & inner

Outer2.java

Why generics?

↳ meant for compilation

↳ type safety @ compile time

↳ no need of explicit downcasting

↳ avoid all class cast exceptions
at compile time

example → Holder.java

day 12 → whiteboard memory d.

↳ < T >

↳ cannot be primitive

day 12.1

↳ no changes at runtime

limitation of array?

→ size change not possible

→ insertion & deletion is difficult

→ not sorted

→ searching 12.1 → complete for 2 min

Test Iterators & or 2

↳ after attaching iterator modification

not allowed → concurrentModExc.

any structural mod

↳ if .hasNext()

↳ if .next() & its .remove() go hand

in hand → IllegalStateException

↳ and only once called

collections-readme → Regarding
Exceptions.

RRRA

Spring Security

- ① WebSecurity Configuration Adapter
- ② Controller

Roles → user → view car
admin → create, delete,
manager → view users only

WebSecurityConfig.java
↳ using → WebAdapter

- ③ Security can be applied based on methods http, post, put, patch

Method level

↳ @so apply at end point level

• antMatchers(HttpMethod.PUT).hasAnyRole("Admin", "Manager")

• antMatchers(HttpMethod.GET, "/users")

• hasAnyRole(" ");

flow to bypass at id level

/userId.

• antMatchers(HttpMethod.GET, "/users/{userId}") .access ("
@UserSecurity.hasUserId(authenticated)
& #userId")

using annotation . -

- ① Controller class → ② Enable Global Method

↳ security(prePostEnabled=true)
to test

@PreAuthorize("hasRole('ROLE_ADMIN')")
@GetMapping("/users")
get all users.
or hasRole('ROLE_MANAGER')

→
@PreAutho("userSec.hasUserId")
@GetMapping("users/{userId}")

Post Authorization

↳ to check before returning

@PreFilter →

@post filter →

④ we can use "annotation"
or

Spring Security

Start

→ @Configuration

public class WebConfiguration
extends WebSecurityConfigurerAdapter {

Java hats

⑧ symbol to represent composition and aggregation in UML diagram

User-a → .

one entity depends on other entity for some functionality

Name Mangling → for overloading

Compiler generates its own names (different)

Why method overloading return type ignored?

because in java collecting return type is not compulsory hence compiler ambiguity which value to return

int a = add(10, 20)

→ here compiler knows what to return

add(10, 20) → don't know

hence return type is ignored

User Details Service

currently logged in user is stored in Authentication object

To configure Security we extend WebSecurityConfigurer Adapter {
 @Configurable}

Method

↳ AuthenticationManager Bean
↳ configurer(AuthenticationManagerBuilder auth)
↳ configurer(HttpSecurity)

① UserDetailsService is abstract given by spring which we implement

② it returns UserDetailsObject

③ Granted authority should be prefixed with ROLE-

List<GrantedAuthority> Authority =
new ArrayList<>();
Authorities.add(new SimpleGrantedAuthority(this.user.getRole()));

How to tell Spring to use our service & User Details @Configuration

→
 @protected void config(AMBuilder auth)
 auth.userDetailsService(userDetailsService)
 .passEncoder()
 @bean

from PanEncoder Bean factors

Spring Security

④ spring security is implemented using filters

→ filters → Delegating Filter

Proxy chain

→ controllers

↓
Security filter
chain

Authorised

↓
Authentication filter

yes

Principal

↓
Authentication
Manager

stores
currently logged
in user

↓
Authentication
provider

Authenticatn
Token

{ does
authorisath
authenticatn
check if endpoint
in allowed }

How to tell to ignore some path?

@Configuration

```
protected void configure(WebSecurity web)  
    WebIgnoring()  
        .antMatchers("/users/**")
```

→ add
preAuthorize (
"hasRole('ROLE-ADMIN')"
or "hasRole('ROLE-MANAGER')")

Role Based Authorisation

① Userinfo Details

② Web Security Config.

using annotation

① @EnableGlobalMethodSecurity (

PrePostEnabled = true)

→ over working fine

JWT Token

→ token based authorisatn

- header
- payload
- body

④ dont store any

→ pom.xml

→ jwt (jsonwebtoken)

→ jaxb-api

↳ java.xml.bind
↳ jaxb-api

↳ jwt

0.9.1

utility →

↳ JWTUtility

↳ @Serializable, @Component

① generate token (UserDetails)

② doGenerate

③ isToken

→ Authentic API → Return

Token

→ User Name → ↗
password → ↗

→ DTO → JWT Request → private String

Res.

username,
password

JWT Response → String JWTtoken

→ controller

→ authentication API

@Authenticat@Authenticate (JWTRequest request)

{

↳ Return token

④ authentication manager → @Bean

↳ to authenticate user

→ JWTUtility → Secret

④ Value ("\$\$jwt.secret")

private String secretkey;

↳ in application.properties

JWT:

secret = secret123

Authenticating token

filter → jwt filter → extend

On per requ.

@Component

disabled session manager

Spring React integration

MVN spring-boot:run

→ search functionality

→ on key Up = "search")

const = () => {

let filter =

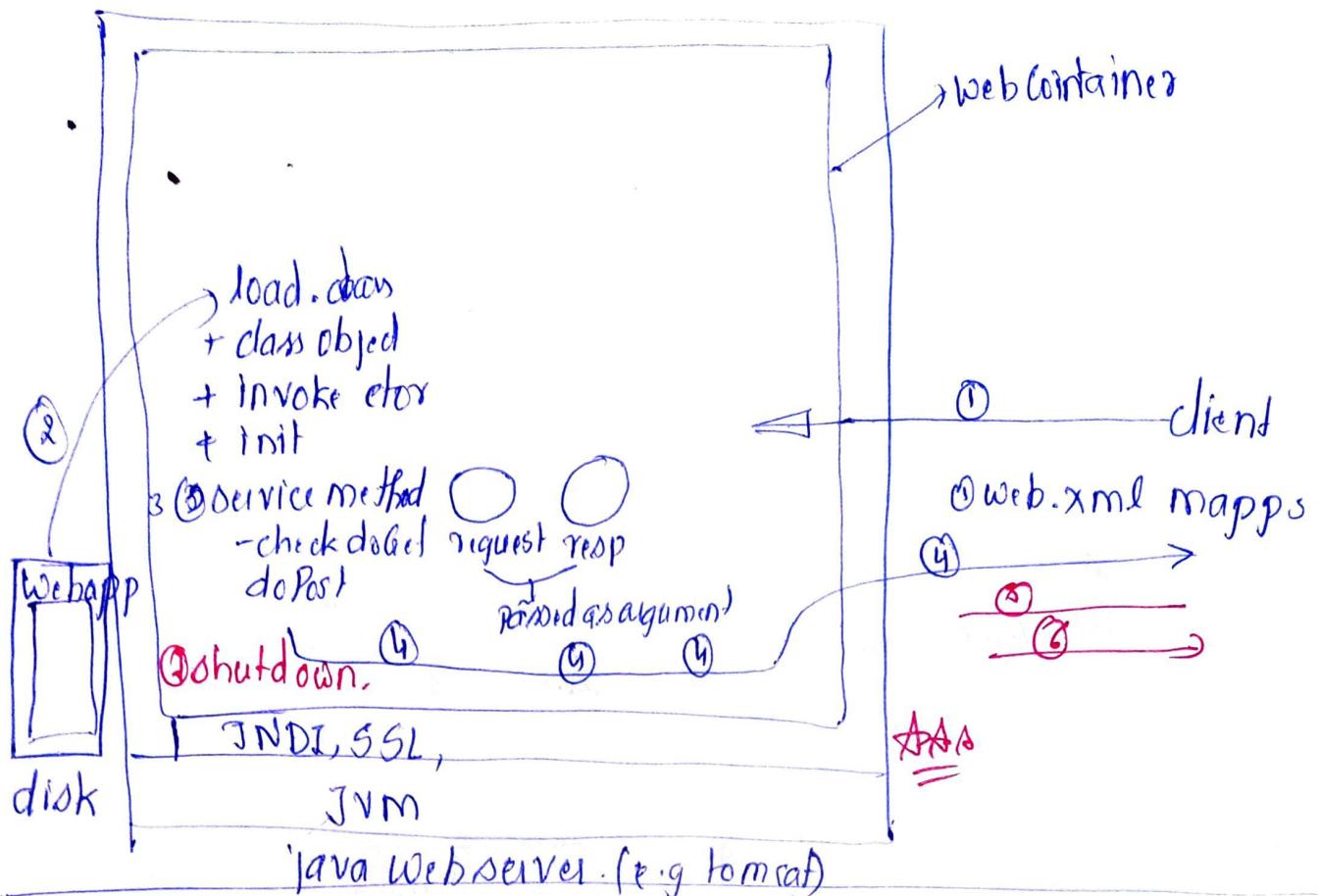
map(e => { if (e.firstName.toLowerCase()

• includes (searchTerm.toLowerCase())

Advantages Java based

Servlet life cycle?

→ servlets are abstracted



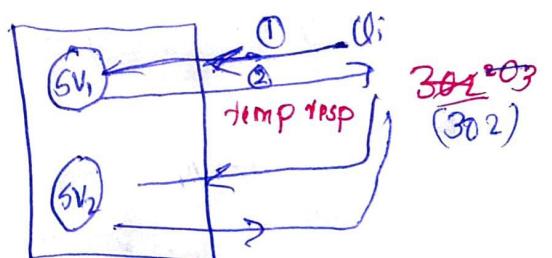
Java Web Server (e.g. tomcat)

```

web.xml-> <servlet>
          <servlet-name>
          <class>
          <load-on-startup>1
</>           > pages loading
<servlet-mapp>
  <servlet-name>
  <url-pattern>.
</> .
  
```

Navigatn in servl

- ① client pull
`a href="#"`
- ② client pull 2
req. sendRedirect();
- ③ rd.forward();
rd.include();



http Redirection

Spring → View name
"redirect: url"



{class::method}

• Collect (Collectors.toList());

Reduce

Reduce(0, (int a, int b) → a+b)

$\sum = \sum + a + b$, equivalent

sum=0 initial

InSummary Statistics Summary

= IntStream.of(7, 13, 19, ..)

• summaryStatistics

count, min, max,

Regular Expression

To match character →

Pattern pt = Pattern.compile()

String regex = "java lesson"

→

.. → 2 char

... → 3 char

↳ only 3 are allowed
on

abc ✓

ab X

\\w\\w\\w → (a-z, A-Z,
abc ✓ 0-9, -)

\\w

"of," → false.

↳ its not wordchar.

321 - 321 ↳ only one positn

regular expression, String methods

"\\w" ↳ wordchar

"R%" ↳ \\w\\w → true
↳ capital

\\w\\W ↳ explanation

"R%l" → false

"R%_l" → true

white space → \\s

" " → false

"—" → true

Wd \\D
1% ✓ ↳ not a digit
11 false

e.g.

To escape character → \. \\

• → space also matched

\\d

↳ matches complete white space.
line

^ → start

\$ → end with

Search phone number

321 - 555 - 4321

\\d\\d\\d, ↳ \\d\\d\\d. \\d\\d\\d\\d.

[.] ↳ escape not required.

[.*\$] [] { }

800-

900 -

[89] 00E-0] \d \d

[8-9] _

↑ range

Quantifiers →

* → 0 or more

? → 0 or 1

+ → 1 or more

{3} → Exact

{3,43} → (min, max)

& 900-333-4232

\d{3} * \d{3} * \d{4}

Max. ? \s Mr.

i
0 or 1
=

group → ()

m(r | s)rs

email matching

[a-zA-Z]+ @ [a-zA-Z]+ . com

ddeepak8894@gmail.com

+ve lookahead

x (? = y) x if followed by y

x (? ! y) x if not followed by y

a (x <= y) x

(x <= y) x x if after y

(x < ! y) x not y

Password matching

Ddeepak7794@

(? = . * [a-zA-Z])(? = . * [A-Z])

(? = . * [0-9])(? = . * [\\$\@])

[a-zA-Z0-9@\\$] {8,3}

aa aA

+ve lookahead (Right side)

(? = . * [a-zA-Z])

-ve lookahead

(? <= y) x

→ groups, +ve lookahead

Core-Java

What is wrapper class and what is its use?

Why only one public class allowed in one .java file?

→ because it is Java language specification

→ it is done so because

Pkg A

↳ java

↳ class C
↳ class D

the way compiler
Pkg B reader
↳ java for class
files

↳ ref class

to get ref in class java compil.
Search for C.class → C.java

Compil. ← Search
all C.class

load One .class file per class

Pkg A

↳ C.java

↳ publication C

Pkg B

↳ D.java

↳ publication D

C

① C.class → C.java → compile
& link it to reference

How can we pass argument

to method by reference?

All argument in Java are pass

by value.

→ objects are passed as reference

→ but reference are passed as value

To achieve pass by reference

① we use wrapper classes

② we use arrays for that

both class objects always both are reference type in Java created on heap

What are the advantages of generics?

Advantage → type safety which enable use to see compiler error before run if there is class mismatch for e.g.

ArrayList A1 = new ArrayList();

A1.add("A")

A1.add(1)

A1.add("B")

A1.forEach(System.out::println)

for(String s:A1)

System.out.println(s)

We have classcast exception.

If we use < > generic it can be detected at compile time.

disadvantage → Not allowed in overloading

② primitive is not allowed.

③ it is only at compile time

Comparable vs Comparator

compar(T other) compareTo(T O, T O₂)

<p>Natural sort → defined by owner class logic inside <u>UDT</u></p> <p>Custom sort → more than one sorting Criteria is allowed</p> <p><u>What is</u></p> <p><u>comparable vs Comparator</u></p>	<p>Iterator<String> it = new list.iterator(); while(it.hasNext()) { String ele = it.next(); }</p>
<p>→ java.lang.</p> <p>→ implemented in UDT class</p> <p>→ its method is CompareTo</p> <p>→ natural sorting</p> <p>→</p>	<p>→ java.util</p> <p>→ implemented by Non-UDT class</p> <p>→ its method is compare</p> <p>custom sorting + multi criteria sorting</p>
<p><u>fail-fast & fail-safe</u></p> <p><u>fail-fast</u> → the Iterators which don't permit structural modification all the iterators on a List are fail-fast any ways if permit structural modification by using <u>it.remove()</u>;</p>	<p>while deserialisation version of <u>serialised object</u> and object loaded in <u>JVM</u> method area if both are compatible then only <u>deserialised</u></p>
<p><u>ConcurrentModificationException</u> is thrown</p> <p>→ It works on original list rather than copy of Array</p>	<p><u>Object slicing</u> →</p> <p>emp e = new mgr(); here compiler only sees whatever is present in e. i.e Whatever is present in super class it can access only what is present in super</p>
<p><u>fail-safe</u> → they permit structural modification moreover, they work on copy of data they are on specialised ds only.</p>	

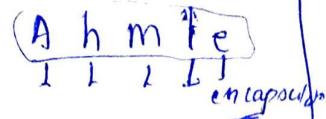
OOP

Abstraction

pillar →

Major pillars

- Abstraction
- hierarchy
- modularity
- Encapsulatn



Minor pillar → Typing

- Concurrency
- persistent

- abstraction → getting essential details of the system
- real life → laptop → we are abstracted from internal hardware & software details

coding Arrays, `toString()`

→ we are not concerned about its implementation.

`list.forEach()`

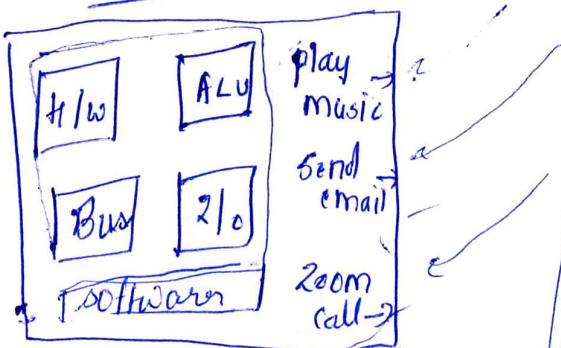
→ we are not real concerned about.

→ constructor

Encapsulation - Binding fields & method together deals to internal to system

abstraction deals with external world how we will communicate from external world

e.g



Hierarchy - Reusability

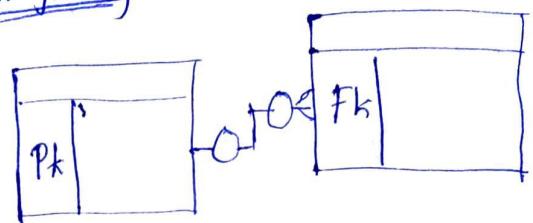
hybrid grains are inherited from non-hybrid grains

Dependancy notation

Dependant -----> Dependency

child → Parent

foreign key



+ one

many

one and only one

zero or one

one or many

zero or many

Aggregation

→ Aggregation

→ Composition

→ Association

→ Inheritance

-----> dependency

Why return type in method overloading is ignored?

`int a = add(a+b)`

`add(a, b)` → compiler will not know what to return, hence return type is not considered in method overloading

servlet life cycle

- ① URL → mapped in web.xml
- ② corresponding class loaded in JVM
- ③ object is created
- ④ constructor is called
- ⑤ init method is called
- ⑥ service → doPost, doGet method is called
- ⑦ shutdown

Status response of http

- ① 100-199 → information responses
- ② 200-299 → success full responses
- ③ 300-399 → Redirection messages
- ④ 400-499 → client error responses

500 - 599 → server error

401 → unauthorized

402 →

403 → forbidden

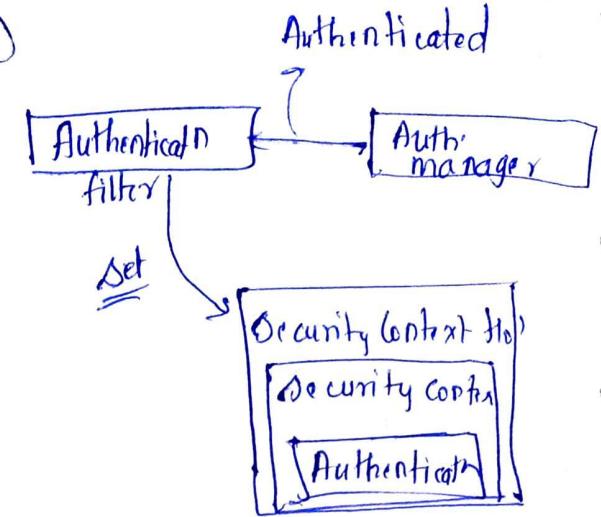
404 → not found

405 → method not allowed

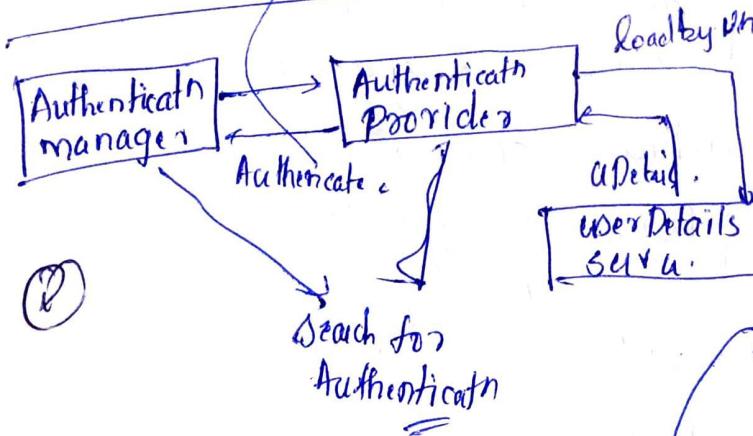
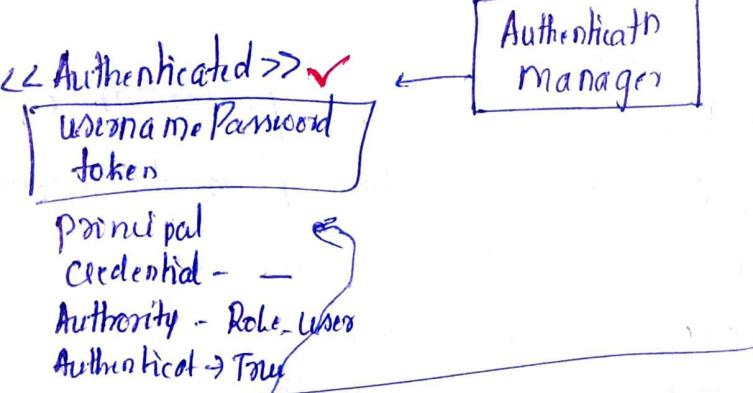
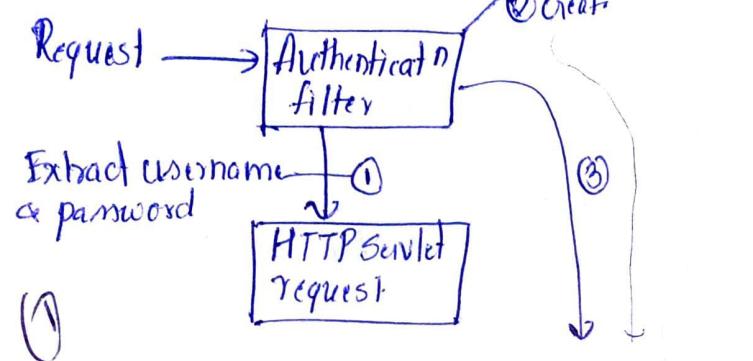
http redirection

- ① res.sendRedirect()
- ② rd.forward()
- ③ rd.include()
- ④ onclick, href, <a>

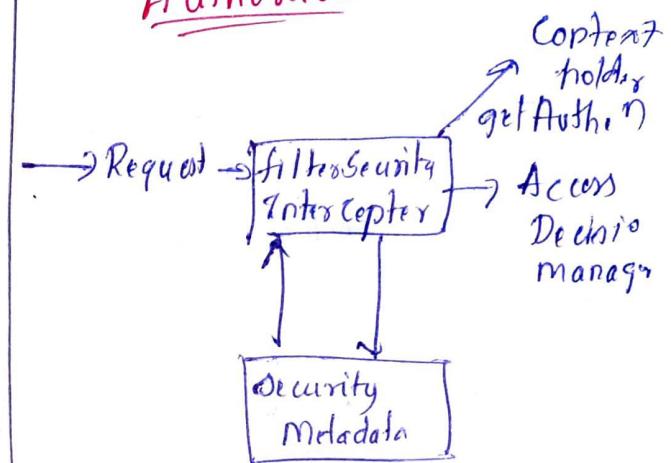
- Why bootstrap?
- auto compatibility with size of screen by using container fluid → Responsive website
- huge amount of attractive components
- easy to include, huge community support
- Ready made templates



Spring Security



Authorisation

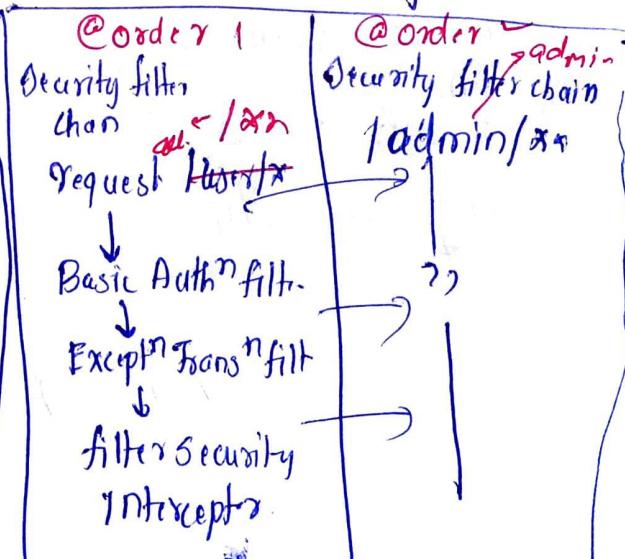


filter chain

@order(1) , @order(2)

first next =

filterchain Proxy



Stream → datasource → intermediate
→ termination.

→ count is not effected
→ distinct().count().

→ .anyMatch(a → a.contains("abc"))

filter →

Collection →

Iterable ①

collection ②

List ③

↳ LinkedList ④

↳ ArrayList ⑤

↳ Vector ⑥

Set ⑦

↳ HashSet ⑧

↳ TreeSet ⑨

↳ LinkedHashSet ⑩

Map ⑪

HashMap ⑫

↳ LinkedHashMap ⑬

'java.lang.Iterable' ⑭

iterator

foreach

Collection ⑮

List ⑯

ArrayList ⑰

LinkedList ⑱

Vector ⑲

TreeSet ⑳

Set ⑳

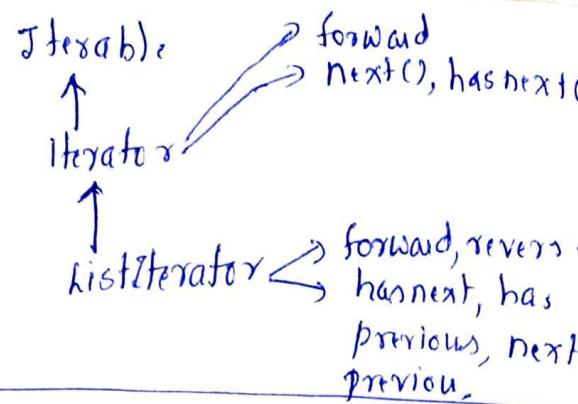
Set ⑳

Set ⑳

Set ⑳

Set ⑳

dnio
dior



MAP < k, v >

HashMap < k, v >

↳ Unordered

↳ Unsorted

↳ Ordered

↳ Unsorted

↳ Sorted

↳ TreeMap

< k, v >

↳ SortedUnorder

Super

↳ Super fn, ln

↳ Super, visible Members

Person → fn, ln,

{ employ → joinDate, & specialisation,

this → non-static

→ own variables, methods

Abstract → Non-concrete methods
are present

→ Variables, constructor,

→ Implementation compulsory
for extending class

→ only declaration

Interface → 100% abstract

→ Data m → public static final

→ Methods → public abstract

5
5
5

→ { 5 5 }

flatmap

d.

Stream<String> newStream =
 details.stream.flatMap(
 d → d.getParts().stream)
 New stream get attached to old
 stream. Is lost

Reducer → Reducer (initValue,
 accumulator function)

all .reduce(23, (a;b) → a+b);
 all : {1,1,13}
 → O/p = 26

Collect (Collectors.toList());

why java?

Hibernate Architecture

Java applicn

↓
hibernate

↓
JDBC

↓
RDBMS

→ session factory

Driver Manager → get Conn → no query
 Connection → statement → ↪ Session

Statement

wrapper



→ session.persist()
 • delete()
 • update()
 • find() or get

Ready made ↗

Dialect: - use for certain
 differences b/w databases
 (it is dependant on mysql,
oracle, ...)

handles difference
 actual Query generat'

① By default auto commit = false

② Transaction compulsory
 for DML

Hibernate Cache

level 1 → session cache → compulsory
 level 2 → session factory

all the changes on cache
 object are always tracked &
 while committing changes
 are transferred to db

Transient object → hibernate
 don't know regarding
 this object → not saved

Persistent object →
 - hibernate
 session cache

Detach → saved to DB
 Tx committed after

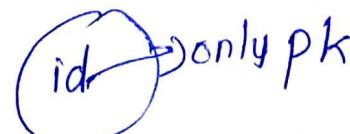
Delete:

- Removed State
- deleted from DB

Get vs load

Returns exact object Book.java.

→ returns



Proxy → hibernate
build object

for id → other than pk → then query

is fired

Lambda → assigning function to variable.

Comparator c = (a, b) → return a.comparator(b)

sort(list, c) =

Syntax must look like
Interface impl.

Anonymous Inner class

Animal a = new Animal()

{

p.v.make();

sys

{;}

object is

Runnable p = new Runnable();

not

cover run();

of Runnable.

rather it's

of implementation

class

Quick sort

7 8 9 10 3 2 5

less 7 more

3 2 5

2 3 5

//

8 9 10

8 9 10

=

7 8 9 10 3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

7 8 9 10

3 2 5

2, 3, 5, 7, 8, 9, 10

time

order & having

group

join delete what

1NF

- ↳ mixed datatypes not allowed
- ↳ ordering as per row order
Not recommended there should be one extra column to do that
- ↳ Table w/o primary key
- ↳ no pk no 1NF

Name	height
A	twenty three
B	22
B	fifteen

no_pk ✗

Name
A
B
C
D

entry ✗ according to height

Name	Data
A	Q3, Q4, Q5
C	One, two, three

✗ non repeating groups

2nd NF

pk

Player-id	item-type	item_qty	Player-rating

player_id → player_rating ✗

player_id & item-type → qty ✓

non pk element must depend on pk & not on part of pk

3rd NF

P.id	P.rating	P.skill lev
A	Beginner	4

Player-id → player skill

P.id → skill → rating

AAA

every non key attribute in table should depend on key the whole key & nothing but key

Arrays to array list

Arrays.asList()

Collection.addAll(list, array)

replaceAll(" ", " ");

Map Vs flat map

one to one

one to many flattening

{1,2},{3,4},{5,6}

→ {1,2,3,4,5,6}

flatMap(stream),

Regular expression

[Hh]ello

(H|h)ello ? → optional

Hello, hello ✓

things not mentioned in resume

→ I love documenting things → git

→ teaching others (3)

→ drawing table

→ I love exploring things = ①

→ youtube videos → make things easy by using all feature

→ bought lab to teach nephew

→ use gitlab repository, commits.

→ Pages → note references, time, stamps

→ docker, selenium,

→ time management (enjoy my work)

farming → how to grow sugarcane,
courgette, onion,管理

Introduce yourself

↳ PGdac → sunbeam

↳ entrance → 25th in grad

↳ 12 → 10th

↳ 10th → CG

Web programming technology

↳ .html → Form

→ Modal

React, Bootstrap → showing information on browser

→ OOP → abstraction

encapsulation

polymorphism }

Inheritance

Model → Blueprint of object.

→ Implementation of object

→

abstraction → essential details

encapsulation → data + method

→ inheritance → code reusability

→ method overriding | method overloading

→ static

→ Java c

→ no restriction

→ same / subclass

→ exc handling

→ ret ignored

→ signature different

→ dynamic polymorphism

→ JVM

→ same / wider

→ subclass -

→ new / wider exception

→ covariant return type

→ same signature

need of database

↳ organised way

↳ DB operations are efficient

happiness

at docked

aus, docker

im

sort, the data

security to data

first

row

JRCITS

DBMS ↳ SQL lang = Structured

→ fixed coln, foreign key, primary

St =

Name of student | student

A	1
B	2
C	3

student | course

1	d
7	e
1	f
2	d
2	e

Select Name, course from students. \$

inner join course on c.student = s.student

Why HMs?

↳ pandemic

↳ patients

↳ lack of coordination b/w staff

↳ availability resource

↳ billing transparency

↳ easy employee management

↳ patient management

select count(emp_id) from employee_id;
 sum(oal);
 max();
 min();

① data transfer from one page to other
 ② spring security
 ③ jwt token
 ④ data flow
 ⑤ docker file → Configuration (optimization)
 → git, spring vs spring boot + ds
 sessions storage & session (" ", !)
 → navigate(" /admin", MVC)
 state: {
 path: user.path

= useLocation()
 location = useLocation();
 const { doctorId } = location.state

MVC X
 git
docker

① git init
 ② rename
 git m main mast

git add.
Commit
 git branch -d (branch)
 git checkout -b second
 git

→ git checkout -b second
 → git branch -d third
 → git checkout main
 → git merge
 git branch -m main
 mast

git br
 git checkout -b second;
 git -m
 → docker image ls -a
 image prune -a
 docker image rm <

→ Binary, insulation, selection,
 3 4 0 1 2
 0 2 1 4
 0 2 1 4 3 5