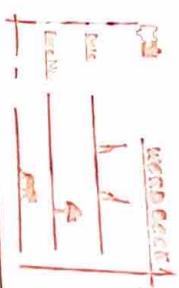


①

Basic Spring Boot



Q.11 open session and current session different?

Ans → open session :- transaction ko manually handle

current session:- transaction ko automatically handle kota hai

Q.12 Service kese generate kte hai?

Ans → service "login" → ng g s login

Q.13 Component kese generate kte hai?

Ans → component "welcome" → ng g c welcome

Q.14 Service classes -

Ans → http Service

Service locator

Auth Service

Q.15 Rest API kesi bnayi he?

Ans → @RestController bnayi he. JSON me data leta hai -

hui to JSON me data leta hai -

Q.16 View kese banaya hai?

Ans → Angular 'Se' view banaya hai.

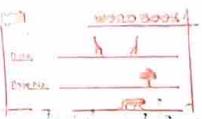
Q.17 Exception ayeji ta kya hogya?

Ans → transaction roll back hogya.

Q.18 Exception propagate kis case me hotha he?

Ans → unchecked exception propagate hota he from se.

(2)



Q.9) Propogate kya se hoti he?

A^s) Service se hi propogate hoti he.

Q.10) Propogate kaise ke liye kya kiya?

A^s) propagation required kiya.

Q.11) Data kon se format me aata hai?

A^s) Data JSON ke format me aata hai.

Q.12) Transaction kaise handle kiya?

A^s) @Transactional

Q.13) Transaction kya handle kiya?

A^s) Service class par kiya hai.

Q.14) pre handle and post handle kb call hogi?

A^s) Request gya to prehandle call hogi

Response ayaega to post handle call hogi.

Transaction handling:

Q.15) Spring Security ke liye kon se module use kiya?

A^s) AOP

Q.16) ORS Response ki method kon si he?

A^s) input error.

Data n

message.

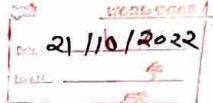
Q.17) What is the functionality of IOC?

A^s) It manage life cycle bean.

Dependency injection of Bean.

(3)

Spring / Spring Boot



Project → 10

Q.1) Use Technologies →

A^s) Jasper Report v - 6.11.0

Log4J v - 1.2.17

J Jwt v - 0.9.1

My SQL-Connector v - 5.1.6

C3P0 v - 0.9.12

Q.2) full forms →

A^s) JPA → Java Persistence API

API → Application programming interface

JSON → Java Script object Notation

AOP → Aspect Oriented Programming

AJAX → Asynchronous Java Script and XML

Rest → Representational State Transfer

JWT → Java Web Token

Q.3) DAO, DTO, Service, Controller

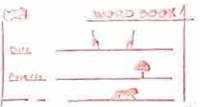
A^s) DAO → @Repository

DTO → @Entity

@Table

@Column

(4)



Service → @ Service

CH → @ Controller / @ Rest Controller
Spring Boot

Q.4) What is controller? How you make CH?
Ans → Controller is a Java class that contains navigation logic by @ Rest Controller.

Q.5) What is service?

Ans → Service performs business operations with the help of DAO and handle Transactions.

Q.6) What is DAO?

Ans → Data Access Object. Data transfer one layer to another layer. DAO implemented in ORM or JDBC and perform crud operations in Database.

Q.7) What is DTO?

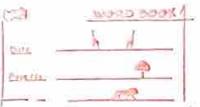
Ans → Data transfer object that containing multiple attributes.

Q.8) What is non business primary key?

Ans → Non Business primary key which doesn't relate with our business data.

Q.10) Why we use non business primary key?

(5)



Ans → Non Business primary key to maintain referential integrity.

Q.11) What is AOP?

Ans → Aspect oriented programming enable modularization such as transaction management that work across multiple types and objects.

Q.12) Key element of AOP?

Ans → (1) Join point :-

(2) Advice :-

(3) Point cut:-

Q.13) What is Model?

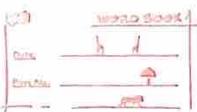
Ans → It is a map object that contains data in the form of key value pairs and carries data from CH to view.

Q.14) What is Spring?

Ans → Spring is a light weight container which maintains the life cycle of Bean.

Q.15) How do you apply Jasper which tool are you using?

⑥



Ans → making JasperIT and Using the classes of Jasper and use tool: Tibco Jasper soft Studio.

Q.16) What is Data Binding and how does Data Binding?

Ans → Bind Request parameters and method arguments
① @RequestParam

Bind Request parameters to a bean object and Store into model object

② @ModelAttribute

Q.17) controller class को bind करने and URL को bind करने के लिए कौन सी annotation का use करा?

Ans → controller class ko Bind Karne ke liye, ya URL ko Bind karne hoto

③ @Request mapping

Q.18) How many types of validation?

Ans → two types of validation.

- ① programmatic.
- ② declarative.

⑦



Q.19) you apply in your application?

Ans → Declarative: input and Business

Q.20) How you implement Spring in your application?

Ans → adding Spring dependency.

- ① Spring core
- ② Spring context
- ③ Spring orm
- ④ Spring web
- ⑤ Spring web mvc

Q.21) How bean is configure?

Ans →

Q.22) How many types of container?

Ans → There are two types of container.

① Bean factory container.

② Application context container

Q.23) what is Bean factory container?

Ans → Bean factory container is light weight container and used for enterprise (desktop) application.

Find package (org.springframework.beans.factory).

Bean factory).

(8)

Q.24) What is Application Context Container?

Ans → Application context container is a heavy weight container used for web application it is the child interface of Bean factory container.

Q.25) Which container you used in your Application.

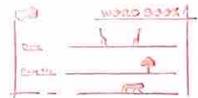
Ans → Application context container.

Q.26) Why you use Application context Container?

Ans → It is used for web application and it provide all the facility of Bean factory and also some additional facility its own.

Q.27) How many key implementation classes of Application context container.

Ans → There are 3 key implementation classes of Application context container.



(9)

Q.28) What is IOC?

Ans → Inversion of Control Provides DI (Dependency Injection).

Two way of dependency injection.

- (1) Constructor, called constructor injection.
- (2) Setter method, called setter injection.

Q.29) Why use DI? Advantage of DI?

Ans →

Q.30) What is Row Mapper?

Ans → It is an interface. JDBC Template uses this to map a row of Result Set to an object.

Q.31) What is contextual session?

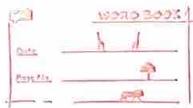
Ans → Hibernate itself manages one current session per transaction in contextual session.

Q.32) What is current session?

Ans → Current Session which is bound to the life cycle of transaction and will be automatically closed and transaction ends.



10



Q. 33) What is Dispatcher Servlet?

Ans → It is main servlet that handles all the HTTP requests and response.

Q. 34) How can we forward request?

Ans → Using prefix return "forward:/ userList" in ctI (by default forward)

Q. 35) How can we Redirect Request?

Ans → Using prefix return "Redirect:/ userList" in ctI

Q. 36) Setter Injection, Constructor Injection.

Ans → ① Setter Injection:- In Setter Injection Dependant Bean is injected by setter method and with the help of <property> tag by Using "ref" attribute

② Constructor Injection:- Constructor Injection is done by Construction of Bean by Using <constructor> inside <bean> tag

Q. 37) XML configuration

Ans → XML configuration:-

Auto wiring can be enabled in XML configuration by Auto wire attribute inside bean tag.

Ex → <bean id="User" class="in.co.rays.user" autowire="byname" />

11



Q. 36) form se parameters get kese karege?

Ans → @request param

Q. 37) URI se parameters get kong he to kese karege?

Ans → @path variable.

Q. 38) AJAX calling kaise karege?

Ans → HTTP Client ki get & post method se call kiyा hai.

Q. 39) Modules we use?

Ans → Spring core

Spring context

Spring orm

Spring daa

Spring aop

Spring web

Spring web mvc

Q. 40) JPA ki methods?

Ans → Persist → void Return karta he

merge,

save → Serializable Return karta he

Q. 41) ioc kya hota he kya data hai follow kaise kiyा hai?

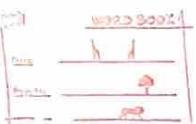
Ans → inversion of control

auto wire kiske follow kiyा hai.

Q. 42) Session faceted design pattern kaise follow kiyा?

Ans → DAO ko auto wire kiyा hai service pe.

(12)



Q.43) Call Back function aur **जैसा क्या कहे जाएगा** use किया?

Ans → call back function is a function a function which is called in a function is a parameter.

Q.44) Which Architecture you used in your project?

Ans → Layered Architecture.

Q.45) How many layer in this architecture?

Ans → There are 4 layer

- ① Presentation layer.
- ② Business layer.
- ③ Data layer.
- ④ Integration layer.

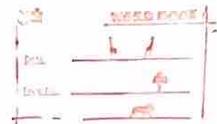
Q.46) Which module in applied at each layer of project Spring?

- Ans →
- | | |
|----------------------|--------------------------------|
| ① Presentation layer | → Angular |
| ② Business layer | → Spring AOP |
| ③ Data Access layer | → Spring DAO, Spring ORM |
| ④ Integration layer | → Spring mail, Spring Security |

Q.47) Kon si layer pr kon sa logic apply kiyा?

- Ans →
- ① presentation layer → view logic; navigation logic.
 - ② Business layer → Business logic
 - ③ Data Access layer → Data Access logic
 - ④ Integration layer → Integration logic

(13)



Q.48) Kon se logic kya apply kiyा?

Ans →

- ① Presentation logic → (JSP) (Angular)
- ② navigation logic → (Controller)
- ③ Business logic → (Service)
- ④ Data access logic → (DAO class)

Q.49) Which design pattern is follow by your application?

Ans → There are seven design pattern in use your application

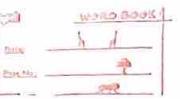
- ① front CH
- ② Session facade
- ③ DAO
- ④ DTO
- ⑤ Builder
- ⑥ Bridge
- ⑦ command.

Q.50) Kon si layer pr kon se Design pattern use kiyा?

Ans →

- ① Presentation layer → front CH
command
DTO
Builder
- ② Business layer → Session facade
DTO
Bridge

(14)



- ③ Data Access layer → DAO
DTO
Bridge.

- ④ Integration layer → DTO
Builder

Q. 51) What is session facade Design Pattern, In which class you apply this design pattern?
Ans → Single entry point for entire application.
Apply in service class.

Q. 52) What is DAO, DTO, frontCH, Bridge Design pattern ?
Ans → front CH:-

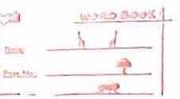
DAO :- Data Access object transfers data one layer to another layer.

DTO :- Data transfer object containing multiple attributes.

Bridge :- Single interface multiple implementation

Q. 53) How is Bridge Design pattern how do you apply in your application?
Ans → Single interface and multiple implementation class

(15)



Q. 54) What is front CH design pattern how did you apply in your application?

Ans →

Q. 55) What is Auto wiring?

Ans → It enables implicit bean dependency injection.
Autowiring can be done by two ways
① XML configuration
② Annotations

Annotations can be apply attributes, Setter method
Constructor.

Q. 56) Why use @ autowire annotation?

Ans → @autowire annotation is used on class them
class inject all dependency.

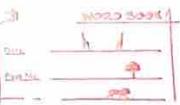
Autowired (required = false) what it means?

Ans → It means mandatory, if it is true then it is throw exception.

Q. 58) Auto wired modes define?

Ans → There are 4 modes of autowiring.

(16)



- (1) auto-wire "no" →
- (2) auto-wire "by name" → by name autowiring is done by property name
- (3) auto-wire "by type" → by type autowiring is done by property type.
- (4) auto-wire "constructor" →

Q.59) What is auto-wire candidate = "false" ?
Ans → means bean can't be inject in another Bean

Q.60) Annotated classes are scanned by which tag?

How you enable annotation based configuration?
Ans → <context: component-scan base-package = "in.co.rays.project10.*"/>

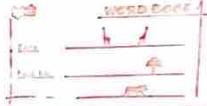
Q.61) How many ways to apply transactions in Service class?

Ans → XML configuration
@Transactional annotations

Q.62) Define attributes of transaction?

Ans →

(17)



- (1) propagation. Required :-
- (2) Propagation. Required. new :-
- (3) Propagation. Supported :-
- (4) Propagation. Not Supported :-

Q.63) Bean factory vs Application context difference?

Ans → Bean factory :- Does not support the annotations based dependency injection.

Application context :- Supported Annotations based dependency injection.
@Autowired

Q.64) Tools Use :-

- (1) Eclipse
- (2) VS code
- (3) SQLyog
- (4) ERwin
- (5) Post man
- (6) Ergo UML

JPA kimethods

- (1) persist →
 - (2) merge →
 - (3) remove →
 - (4) find → -
 - (5) create Query → Typed Query / Object Create
- Kati ne

Typed query (T? query = Entity manager.createQuery()

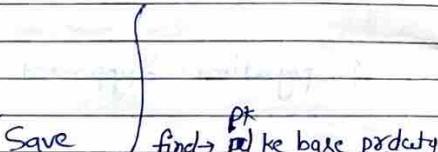
(18)

Q. 65) Display logic and Submit logic kya likhe he?

Aus → Display logic → @GetMapping
 Submit logic → @PostMapping

Q. 66) JPA Methods?

Aus → ① add - persist →
 ② update - merge →
 ③ Delete - remove →
 ④ search - Search →



Q. 67) Startup file (OAS project Application.java) me kon se annotation lgaya he?

Aus → @SpringBootApplication

[
 @EnableAutoConfiguration
 @Configuration
 @ComponentScan]

Q. 68) Id (primary key) kaise bnayi?

Aus → Base DTO me
 @Id
 @GeneratedValue
 @GenericGenerator
 @Column

protected long id;

Q. 69) How to fetch data from URI?

Aus → @PathVariable

(19)

Q. 70) JPA Kaise configure kiya?

Aus → Pom.xml me dependency add ki

① Spring Boot - Starter - data - JPA

Application.properties

① Spring.jpa.show-sql = true

② Spring.jpa.properties.hibernate.format-sql = true

③ Spring.jpa.hibernate.ddl-auto = update

④ Spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.

→ DAO class me entity manager ka object bnaya hai [Base DAO Mai]

→ Persistent-Context ko auto wire karte hai.

Q. 71) Dcp. Kaise configure kiya?

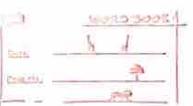
Aus → Pom.xml add dependency
 ① MySQL
 ② c3po

Application.properties

① Spring.dataSource.driver-class-name = com.mysql.cj.jdbc.Driver

② Spring.dataSource.url = jdbc:mysql://localhost:3306/project10

(20)



(3) Spring.datasource.username = root

(4) Spring.datasource.password = root

Q. what are the key features of spring DAO?

Ans → There are three key features of Spring DAO

- (1) Resource management
- (2) Exception Translation
- (3) Transaction Handling

(1) Resource management:- programmer does not need to open or close database connection.

* Spring DAO implicitly open & close database connection

(2) Exception Translation:- * Spring DAO translates all native JDBC and ODM exceptions into wrapped exception.

* Data Access exception is propagated to Service layer

* programmers do not need to handle database exception.

* Service classes handle database exception.

(3) Transaction handling:-

* Service classes handles transaction with the help of Spring AOP

(21)

Q. front hand to Back hand convert?
front hand:-

Http client module का import किया है व उसकी get and post method का use किया है api वाले करने के लिए

Front

Back Hand:- अप्सें प्रेट मैन फाइल का use किया जाता है

With the help of Cors
Cross origin Resource Sharing

(1) In start up file
cors configurer () में,

→ web mvc configurer Adapter का object बनाये हों
web mvc configurer w = new batch mvc configurer();

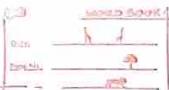
(2) web mvc configurer ki add cors mapping का use किया हो।
aur uske andar cors Registry का object registry pass किया हो।

→ add cors mapping (cors Registry, Registry)

(3) cors Registration का object banaya (cors) & cors Registry ki add mapping () का use किया हो।

→ cors Registration cors = Registry.add mapping ("/*")

(22)



Q. CORS Registration ki allowed origin ko kaise kiji
hai

Ans → cors.allowedOrigins("http://localhost:4200");

Q. How to make primary key or generate the Id?

Ans → primary key Base DTO class me bniye he

@Id

@GeneratedValue(generator = "ncs_pk")

@GenericGenerator(name = "ncs_pk", strategy = "native")

@Column(name = "ID", unique = true, nullable = false)

protected long id;

Q. How do you inject DCP in JDBC?

Ans → By Using Data Source (DCP in JDBC)

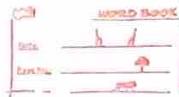
Q. How do you inject DCP in hibernate?

Ans → By Using Session factory (DCP in hibernate)

Q. How do you inject DCP in JPA?

Ans → By Using Entity manager by using @PersistenceContext annotation (DCP in JPA)

(23)



Q. Base DTO ki aur Tarihi ka?

Ans → Abstract classes bnaye Base DTO :-

@MappedSuperClass

public abstract class BaseDTO {

@Id

@GeneratedValue(generator = "ncs_pk")

@GenericGenerator(name = "ncs_pk", strategy = "native")

@Column(name = "ID", unique = true, nullable = false)
protected long id;

attributes liye aaj tak Setter and getter bnaye

→ Attributes:-

protected long id;

String createdBy;

String modifiedBy;

Timestamp createdDateTime;

Timestamp modifiedDateTime;

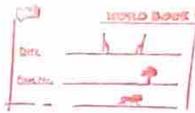
Long org_ID = 0L;

String orgName;

24



25



Q. Base Ctl का क्या है?

Ans → ① public abstract class Base Ctl जो extends Base DAO Int है।

Base form, Base DTO और Base Service Int का क्या है।

② फिर हमने operations लगाए हैं।

- ① op_save, ② op_new ③ op_delete ④ op_cancel ⑤ op_Error
- ⑥ op_Next ⑦ op_Previous ⑧ op_logout ⑨ op_Logout ⑩ op_Set

③ base service को Autowired किया गया है।

④ page size सेट किया गया है।

⑤ फिर हमने कुछ method बनायी हैं।

- ① User context
- ② get
- ③ Delete
- ④ Delete many
- ⑤ Search
- ⑥ Save
- ⑦ validate

Q. Base DAO Interface का क्या है?

Ans → public class interface Base DAO Int < T extends Base DAO > है।

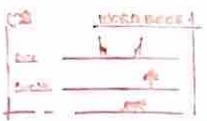
① कुछ method बनायी हैं।

② public long add (T dto, User context user context);
update
delete

③ FindByPK
FindBy Unique key
List Search

List run HQL
Void set Entity manager

26



Q. BASE DAO Implementation में ये क्या किया।

27



Q. Base Form में ये क्या किया।

Ans → Base Form name ऐ ek class bnayi he.
उसकी attribute होंगी वह उसकी setter व getter method
बनाए गए हैं।

- ① id = null;
- ② ids;
- ③ pageNo = 0;
- ④ pageSize = 5;
- ⑤ operation;
- ⑥ createdBy;
- ⑦ modifiedBy;
- ⑧ createdDateTime;
- ⑨ modifiedDateTime;

28



Q. Base Service implementation कि विधि क्या है ?

29



Q. Base Service interface कि विधि क्या है ?

Ans → public interface BaseService Interface < Extends BaseService >

methods होंगे जो define की जाएँ।

- ① add
- ② update
- ③ delete
- ④ save
- ⑤ findById
- ⑥ list Search
- ⑦

11

(30)

Q. User context का उत्तर क्या है?

Ans → User context पर class bna yi he

public class UserContext {

```
private Long userId = 0L;
loginId = "root";
name = null;
defaultRoleId = 0L;
roleName = root;
orgId = 0L;
orgName = root;
orgManagerId = 0L;
```

User Context :- User mention karta he front hand
and back hand ke bich

Q. JWT का उत्तर क्या है?

Ans → JWT Session mention करता है एवं token
generate karta he token base रखता है

Q. Auth Service क्या bna yi he?
Security ke liye bna yi he?

(31)

Q. Auth का उत्तर क्या है?

Ans → login controller Java में controller mapping का है (Auth)

① Request Mapping ("Auth")

LoginController पर front controller apply nahi kiya he

Q. front end पर get and post method क्या करते हैं?

Ans → get:- (end point) का URL दिया जाता है

post:- (URL, formdata), दिया जाता है

Q. entity manager का क्या करता है?

Ans → entity manager JPA की all method provide करता है
and connection bhi provide karta he.

Q.1 front end service class :-

① end point Service :- end point define karte he
Server-URL = "http://localhost:8080";
this.Server-URL = "/Login"

② HTTP Service :- HTTP service me humne custom get and
post method bna yi he
custom bna ke jaani isliye thi kyuki humko
call bhi krenge tha.

(32)



Q. Service locator:- Service locator me HTTP service & end point service ko inject kiya he and Service locator ko all component me inject kiya he.

Q. back end port number change kaise karte he?
Ans → Application properties me Server Port - 8080

Q. HTTP client ki get and post method kya contain kati he?

Ans → bean, form data, endpoint URL contain kati he.

Q. What is modularization?

Ans → Modularization code ko modules me devide karta he.

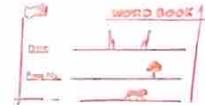
Q. error message CH pr get kaise kiyte?

Ans →

Q. pipe kon se use kiye he?
Ans → (1) toonstate
(2) Date

Q. pipe kya karta he?
Ans → one format to another format me change karta he.

(33)



Q. Custom exception classes kitni banaye he?
Ans → Custom exception classes 3 banaye he.

① Application exception :- Service se.

② Data Base Exception :- DAO se

③ Duplicate Exception :- Khi se bhi Raise ho skti he.

Q. what is User Context?

Ans → User context current User ko hold karta he.
≡ User context back end or front end ke session ko manage or bind karta he communicate karte he.

Q. controller har msg kaise get karta he jo input validation ke hai?

Ans → @ Request Body @ valid Binding Result object

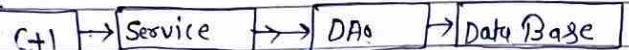
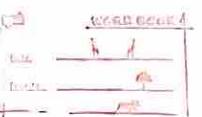
Q. dialect kon se use kiyta?

Ans → JPA ka kiyta he.
MySQL DB53.

Q. DAO classes me kon si exception Raise hoti he?

Ans → DAO class me, ORM specific exception raise hoti he jise ham data access exception karte he. me convert karta he.

(34)



Q. Exception kese bnayi he?

Ans → @Exception Handler se bnayi he.

Q. Session ko Handle kese kiya?

Ans → Jwt Token se session ko handle kiya.
(Backend) par kiya he.

Q. front end pr Data kaise aa Rha he?

Subscribe method se front hand pr data acha he

Q. Observable object ki method kon si he?

Success
fail

Q. Server side me kon sa frame work use kiya

Ans → Spring Boot

Q. Http service kyu bnayi he?

Ans → Http client ke liye bnayi he

Q. list ko iterate karne ke liye kon sa directive use kiya?

Ans → *ngFor

(35)

Input validation Flow:-

Backend:-

- ① LoginCh mai BaseCh ko extend kiya hai form pr hmne validation check kiye he By Using @valid @email @NotEmpty Usme msg pass kiye he (form pe validation set kiya hai)
 - ② Login ch pe OAS Response type ki login method bnayi Usme as a parameter hamne login form our Binding Result ka object binding Result pass kiya.
 - ③ them Base ch ki validate method ko call kiya them Usme Binding Result ka object pass kiya them uska OAS Response ke obj mai hold karayeg.
 - ④ condition check ki (!.res. is Success) & return res 3
 - ⑤ front end:-
- login component ts mai SignIn() method mai resp. result input error mai msg ko get kiya our form msg me hold karwaya them login component.html pe interpolation ki help se particular field pe msg ko print kiya.

(36)

Business validation:-

Front end:-

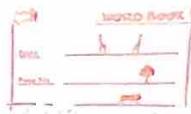
- ① Login Component.html mai Login our password ki field mai button pe click event mai signIn() method ko call kiya hai.
- ② Then Login Component.ts ki signIn() method mai this.httpService.post(endpoint, formdata) aur callback function pass kiya hai.
- ③ this.httpService.post(this.endpoint + "/login", this.form); function (res) {
- ④ Back end:-
- ⑤ Login() ko base chhi se extend kiya hai aur generics mai <UserForm, UserDTO, UserServiceInt>
- ⑥ ORS Response type ki method Login banayi he aur User @ Request Body @valid annotation pass kiya hai aur Binding Result ka object binding Result pass kiya hai.
- ⑦ validate method mai binding result ka object pass kiya hai aur ORS Response ke obj des me hold kiya hai.



(37)

Hindi to English Change:-

- ① Configuration file me (app.module.ts) mai translate package ko import kiya hai.
 - .nem → * translate | ngx-translate / core
- ② Component K constructor mai translate service ko as parameter pass kiya (navbar component).
- ③ Assets folder mai sub-folder banaye (i18n)
- ④ i18n folder mai JSON file banayi he
 - ⑤ Hi.json
 - ⑥ En.json
- ⑦ JSON file mai har ek field me pipe operator ka use karke key value pair me store kiya.
- ⑧ Long parameters ko constructor me pass kiya he.
- ⑨ method - change locale (navbar component.ts) me by default English.



38

Front controller :-

⑪ Front controller class bhyi he extend kiya he Handler Interceptor Adapter Class

Q. 2) Uske do methods prehandle; post handle 'kio override kiya aur usme HttpServlet Request ka object req. As per HttpServlet Response ka object resp. aur object ~~class~~ ka object Handler pass kiya hai.

③ then Starter class me add Interceptors method
bnayi hui usme interceptor registry ka object
Registry pass kiyg hui

④ Registry. Add Interceptor mai (front (+)) pass key
then add Path Patterns mai ("/*"). exclude
path patterns ("/*Auth/*")

~~CORS ENABLE ke se kiyq Flow:-~~

① Startup file mai (project or application).
web mvc Configure type ki cors - configurer
method bazi haiisme

② Usme humne web mvc configure. ka object
brayq brui jisme humne add coops.

٣٩

Q. transaction ko internally handle kona karta he?
Ans → Spring Aop karta he.

Q. Spring DAO and DAO classes me different ?

Spring DAO module he spring provide kota he

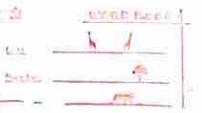
DAO class he manually himne bnayi he Name diya .
he (DAO) Jisse hmko smj me aggaye ki para Access
object Jo Data ko access karegi unhone kuch bhi
ha skta he .

Q. API ko call kese kiya?

Ans → Http client ki get, and post method se API ko call kiya.

Angular पर link कसे बनाए हैं?

(40)



Q. Service ko inject kese karte he?

Ans → constructor (^{http}private: HttpService service);

Q. Service ko define kese karte he?

Ans →

Q. Http Client Module Ko import kese kiya?

Ans → app.module.ts

```

import {HttpClientModule} from '@angular/common/http';
imports ke aagey me HttpClientModule module pass
kiya he.
  
```

Q. Routing kesi ki he?

Ans →

```

const routes: Routes = [
  { path: 'login',
    component: LoginComponent },
    @NgModule({
      imports: [RouterModule.forRoot(routes)]
    })
  ]
  
```

Q. LoginComponent kese laye ho?

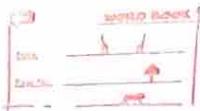
Ans → Import me LoginComponent diya
from login/login.component diya.

Q. Single page Application kese bnayi?

Ans → App Component.html में <routeroutlet> </routeroutlet>
Single page Application bnayi he.

~~index.html~~
Bootstrap

(41)



Q. What is spring Boot?

Ans → Spring Boot is an open source Java-based framework used to create micro service

Q. JSON me data kese aa Rha format?

Ans →

```

{
  "loginId": "rathodardash265@gmail.com",
  "password": "Rathod@1234"
}
  
```

Q. Post man testing kese karte he?

Ans →

post man open करोगे
Body पर click करोगे

मार्केट पर https://localhost:8080/auth/login

get and post चोरे करोगे जो kisi chaliy

Body ke ander do wo ke ander

8

```

{
  "loginId": "rathodardash265@gmail.com",
  "password": "Rathod@1234"
}
  
```

Q. callback function kya karta he?

Ans →

function for response Return karta he.
callback

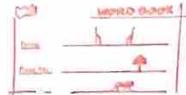
4K



Q. User component एक एक उपयोग केस

- 1 User component. html
 - 2 user component. css
 - 3 User Component. ts
 - 4 User component. specs
 - 5 User list component. html
 - 6 User list component. ts
 - 7 my profile component. html
 - 8 my profile component. ts
 - 9 Change password component. html
 - 10 Change password component. ts
 - 11 User DTO. Java
 - 12 User CH. Java
 - 13 User form. Java
 - 14 User Paint. Java
 - 15 User DAO imp. Java
 - 16 User Service Int. Java
 - 17 User Service imp. Java
 - 18 my profile form. Java
 - 19 Change password form. Java

43



~~Login:-~~

- ① login component. html
 - ② login component. css
 - ③ login component. ts
 - ④ login component - Specs
 - ⑤ forgot password component. Html
 - ⑥ forgot password component. ts
 - ⑦ Signup component. html
 - ⑧ sign up component. ts
 - ⑨ Login form. Java
 - ⑩ login ct. Java
 - ⑪

- Login Component - HTML
- Login Component - CSS
- Login Component - TS
- Login Component - Spec - TS
- Forget Password - HTML
- Forget Password - TS
- Signup Component - HTML
- Signup Component - TS
- Login Form - Java
- Login CH - Java

100

- ① Create maven project and add starter dependencies.
 - ② Use Spring Initializer.
 - ③ Use IDE like STS
 - ④ Spring boot command line interface

Sign up on [this](#) site.

- ① login component →
 - Ans ② login (component . Htm)
 - ③ login (component - C8)
 - ④ login (component Spec) →
 - ⑤ change password Com
 - ⑥ forgot password →
 - ⑦ forgot , spec

(44)

[User list image flow] :-

Q. my profile (image) kese lgayi he user list me?

Ans → UserList (component.html) → pr

① Input type = file liya or change event ka use karte my file ko call kiya.

② my file(s) bnayi he. User component ts me

my file (files: file list) &
this file to upload = file.item(0);
3

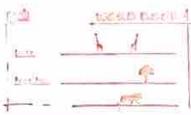
Backend pr →

/user

③ OAS Response type ki upload pic() bnayi he @pathvariable ka path variable ka use karte id ko get kiyा he multipart file ka object bnayi he 'file'
Http servlet request ka object bnayi he request

→ public OAS Response uploadFile (@PathVariable
long userId, @RequestParam
(file) multipartfile file, HttpServletRequest
req);

④ Base Service ki find by Id() ko call kiyा usme User ID or user context ka object pass kiyा or unkо user DTO ke object DTO me hold kiyा.



(45)

→ user prof dto = base service.find by Id (userId,
User (context));

⑤ Attachment DTO ka object bnaya usme description path
or user id pass kiyा.

Attachment DTO doc = new Attachment DTO (file);
doc.setPath (req.getServletPath());
doc.setDescription ("profile picture");
doc.setUserId (userId);

⑥ condition check ki →

if (user DTO.get Image ID () != null)

then

doc.set ID (user DTO.get Image ID());

⑦ Attachment Service ki save method ko call kiyा uske under doc or user context pass kiyा or long me image id hold kiyा

→ long imageId = attachmentService.save
(doc (user context));

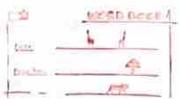
⑧ OAS Response ka object bnayi or OAS Response
ki add Result ko call kiyा usme key
valueroe pair me Image ID ko set
kiyा finally set return response kiyा

→ OAS Response res = new OAS Response ();
res.setSuccess (true);
res.addResult ("ImageId", imageId);



return res;

(46)



- Q. @Controller and @RestController me difference ?
Ans ① @Controller se ham Spring MVC me controller bناte he
@Request mapping
② @RestController ka use Rest API banane ke liye
kiya he.

Two annotation ka combination he
① @Controller
② @Response body

- Q. Spring MVC and Spring Boot me difference ?
Ans

Spring MVC

Spring Boot

- ① Spring MVC ko @Controller se bناte he
② Spring MVC me server manually deploy kرنے pdty
he
③ Spring MVC XML, Tiles, HTML ke form me data Return
karta he
- ④ Spring Boot me already embedded server hota
he
⑤ Spring Boot JSON ke form me data Return
karta he

- Q. Call back function ka use kرنा ?
Ans
Call back function ka use http service ki
get and post method me kرنा he (frontend)

(47)

@Controller → display logic
@PostMapping → submit logic
Payload →ognist payload ki → base list compare. IS

Q. flow of payload ?

Ans → payload :-

Backend →

- ① Respective controller or Response type ki payload bناti he
→ @Net mapping ("Payload")
public ORS Response payload ()
- ② ORS Response ka object bناya
→ ORS Response res = new ORS Response (true)
- ③ Respective DTO ka object bناya DTO
→ Role DTO dto = new Role DTO ()

- ④ Respective service interface ko autowire kرنा
@Autowire

RoleServiceInt roleService = null;

- ⑤ Respective Service ki search () ko call kرنा usme
DTO pass kرنा or list ke object me hold
kرنा
- List < dropdown list > list = roleService.search (dto,
UserContent);

- ⑥ ORS Response ki add Result () me key or value
ke pair me data save kرنा
- res.addResult ("role list", list);
then
Return res;

(48)

front end:-

- ⑦ Select tag me option tag liya he-

<Select [(ngModel)]="form.searchParams.roleId">

<option *ngFor="let e of form.payload.roleList" [value]="e.key" && e.value > 33</option>

<select> point liya,

link →

repeat →

<select name="roleId" [(ngModel)]="form.searchParams.roleId"

<option *ngFor="let m of form.payload.roleList" [value]="e.key" && e.value > 33</option>

Router list component ts me base list ko extend
kiygi uski ny constructor thi- DaoFactory call kiygi

<q class="dropdown-item" [routerLink]="/user">

<q class="dropdown-item" [routerLink]="/user">

<q class="dropdown-item" [routerLink]="/user">

payload → base component

[value] = "e.key"

&& e.value > 33

<q class="dropdown-item" [routerLink]="/user"/>

this.servicelocator.httpClient.get(apiUrl + 'm')

→ Self, api: payload function

(49)

Q. List flow :-

Ans → front end:-

- ① nav bar. component.html me dropdown me link bniyi
he jisme "/userlist" or [routerLink] = "[user]" pass
kiya.

Backend:

- ② User CH extends Base CH

- ③ Base CH ki search () call kiygi

- ④ ORG Response ka object bniyi Response.add Data
ko call kiygi usme hmne Base service ki search
ko call kiygi parameter pass kiygi (dto, page No,
page size, user context)

- ⑤ Base service imp me extends kiygi Base DAO

- ⑥ Base service me Base DAO ki search ko call
kiya.

- ⑦ Base DAO Imp extends Base DTO

- ⑧ Base DAO ki get where clause () method ko
override kiygi usme parameter pass kiygi.

(T DTO, CriteriaBuilder builder, Root<T> qRoot);

- ⑨ Typed Query ka object bniyi

- Typed Query<T> query = createCriteria(dto, user context);

- ⑩ pagination handle kiygi.

if(pagesize > 0)

query.setFirstResult (pageNo * pageSize)

query.setMaxResults (pageSize);

(50)



(11)

query ki get Result list ko call kiyq or list ke object me hold kiyq.

→ List^{list} = query.get Result list();

(12)

return list ki

Again front end:-

(13)

form ^{me list ke} to proxy me hold kiyq.

(14)

component.html pr *ngfor loop lgayq

[*ngfor] = "let m for form.list >

(15)

Shni fields ko print kar liyq

m.image id

m.name

m.login id

form me list ke array me hold kiyq

Component.html me *ngfor loop lgayq.

*ngfor = "let m of form.list ;

<td> {{m.firstName}}</td>

<td> {{m.lastName}}

Repeat

(51)

Q. JWT Security / Auth service ?

Ans → Pom.xml me dependency add ki:

①

JWT

②

spring Email Security

① ek class bnayi he web security config usme hmne @Enable web Security annotations ka use kiyq fir hmne web security config adapter class ko extends kiyq

② Http Security ka object Security bnaya he.

③ Session ko stateless kiyq he.

④ every user Request pr hmne token ko valid kiyq he.

⑤ ek or class bnayi he hmne Jwt Authentication entry point usme hmne Jwt Authentication entry point adapter ko extends kiyq.

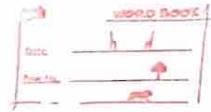
⑥ commence method ~~to~~ use kiyq hai usme HttpServletRequest, HttpServletResponse, Authentication pass kiyq.

⑦ Response ki send error method me hmne your Session has been expired pass kiyq them Response return kiyq.

⑧ JwtUtil token se token pass kiyq he → Token Return karao.

- ④ JWT Token Util class bnayi he usme methods bnayi he
get user name, issue date, expiry date, validity methods bnayi he
then do generate token mai token ko generate kaise
return karvaya.

(53)



Q. Search field / input field ka flow ?

Ay → front end :-

- ① HTML pr ek input field bnayega input type text liya he or [ngmodel] me form. search pograms me first name pass kiya.

- ② click event pr Submit method ko call kiya he.

- ③ component. ts me submit method bnayi he

Service locator. http. service. post(this. api. search,
this. form. Data)

Back end :-

- ④ Respective CH pr extends Kiya Base CH KO
or Base CH ki search method ko
call kiya he

- ⑤ Base CH me OAS Response type ki
Search method bnayi he

- ⑥ OAS Response ka object bnaya ORS Response
or USki add data ko call kiya uske ander
Service ki search ko call kiya or parameter
pass kiye.

(DTO, page No, page size, user context)

~~base service imp extend base project~~ KO

- ⑦ Base service me Base DAO ki search
ko call kiya

- ⑧ Respective DAO imp. me extends Kiya Base
DAO imp class ko or USKI get where
clause method ko override kiya or
parameter pass kiye
(DTO, CriteriaBuilder, Root<T> qRoot);

- ⑨ List ka object bnaya condition check ki

```
if (! isEmptyString ( DTO. getName () ) ->  
    where  
    condition. add: ' ( builder. like ( qRoot. getName (),  
    DTO. getName () + "%") );
```

then

Return list KO

front end :-

- ⑩ Select tag me option tag set ~~list~~

<Select [(ng model)] <form. search. pograms roleId>

option ng for = let e. of form.

*ng for = let m for form. list se print
kiya.

<td> {{ m.firstName }} </td>

⑪ qRoot ka object →

Root<T> qRoot = cg. form (getDTO (class));

(54)

Edit:-



User list component .HTML → pr click event me forward method ko call kiya usme hmne ("User", m.id) pass kiya.

User Component.ts me extend kiya BaseCH
ko

Base CH ki ngOninit() से method ko call
kiya

condition if (form.data.id > 0) से to hmne
this.display method ko call kiya

HttpClient.get (Uri, function (x08))
Uri = http://localhost:8080/User/get \${id}

Backend:-

Base CH me ORS Response type ki get method
bnayi he (@pathvariable, long id) से pass ki

Base Service ki find By Id ko call kiya
Usme (id, User context) pass ki

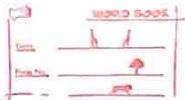
Base Service ki find By PK ko call kiya
Usme (PK, User context) pass ki

Entity manager ki find method ko wj find
(get DTO class.(), pk) pass ki

Return DTO,

(55)

Hindi to English translate :-



- Internationalization:-
- ① ngx translate
 - ② app.module.ts mai translate module ko import kiya

Component ke constructor mai translate service
ko as parameter pass kiya hai
navbar component.ts me

assets folder me sub folder bnaye I18N folder
bnaya cur usme do file bnayi he Hi.json
and En.json

pipe directive : K through translate kring hai
and interpolation K through msg show kiya
hai.

By Default english.

method - change locale (navbar component ts) me
by Default english.

Q. HttpService get and post method code

Ans) get (endpoint, callback) {
return this.get (Httpclient.get (endpoint)).subscribe
(data) => {

console.log ('Data :: ' + data);
callback (data);

3

58

`post(endpoint, beam, callback) {
 return this.HttpClient.post(endpoint).subscribe(
 (data) => {`

`console.log(`Data`);
 callback(data);`

Q. BaseCH ki validate method me kya kiya?
Ans BaseCH me ORS response type ki validate
 method bnayi he usne humne Binding Result +
 object pass kiya he.

② ORS Response ka object bnaya he

③ if (bindingResult.hasErrors()) {
 res. setSuccess(false);

④ res. addInputErrors(errors);

3

⑤ return res;



59

Q. Root ka obj?

Ans `Root <-- q$root = cq.from(getD70Class());`

Q. Request body kya kota he?

Ans Request ko method parameter ke Object se bind
 kota he.

Q. foronCH to live frontend po kya kiya?

Ans `isLogout()` ek method bnayi he.

① localStorage.get('fname')

```
if (JSESSIONID == "null" || JSESSIONID == null)  

  28 this.router.url != "/Auth"  

  28 this.router.url != "/logout"  

    != "/forgot Password"  

    != "/Signup"
```

isme se kuch nhi mila to ~~Router~~ ^{route to navigate} false ~~get call~~
~~Router~~.navigate(['/login']) ^{logi}

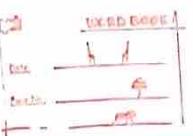
or isme se kuch mila to iste get method
 coul hogi

```
If (this.isLogout()) {  

  console.log('inside isLogout() return true');  

  return true;
```

58



Q. Back end pr Base file kon si bnayi he?

~~Aus → Base C.H.-Java~~

- ② Base DAO imp. Java
 - ③ Base DAO int. Java
 - ④ Base Service imp. Java
 - ⑤ Base Service int. Java
 - ⑥ Base DTO. Java
 - ⑦ Base form. Java

Base Dto-Java

- Base form. Java
 - Base CT. Java
 - Base DAO imp. Java
 - Base DAO int. Java
 - Base Service imp. Java
 - Base Service int. Java

59



~~throws~~

Q. Thorpe ke liye kon kon si class banayi hے?

An → Attachment PTO

- ② Attachment Service int
 - ③ Attachment Service imp
 - ④ User CT
 - ⑤ Attachment DAO. imp. Java
 - ⑥ Attachment DAO int - Java
 - ⑦ Attach Cff. Java, form - Java
 - ① Attachment DTO
 - ② Attachment service int - Java
 - ③ Attachment Service imp. Java
 - ④ Attachment DAO imp. Java
 - ⑤ Attachment DAO int - Java
 - ⑥ User CT

Q. front end पर Base class को सिंचायि हो ?

Ans 2) base Component - ts
② base list Component - ts
③

Q. Single page Application kya hota he?

Ans Single page Application ek hi page load karne ke part of pages dynamic update hone ke ajax calling ke thoray

Q. @Model attribute kya karta he? (Spring)

Ans → (Model) Attribute base phic cell based he.

⑨ Response body 'kyo koto he?'

Ans- Object ko JSON object me converted, karta he

Q. - Apposist Context :-

As JPA ko configure karta he entity manager ka Object Return karta he.

Q. Request Param & Request Body difference

Ans) Request Param:- form se particular field ka Data le kar atya he

Request body:- pure form ko lekar atd
he yq pure form ke data
ko le kar atd he.

(60) Optimal means → handle krna hoto kr skte hain to roll back
nhi kr skte he

Q. Checked exception and Unchecked exception me different?

Ans - Checked mandatory to handle or propagate.
Unchecked optional to handle or propagate.

Q. login ct pr front ct apply kyu nahi ho rhe?

Ans → path get kiya he

path request.get servletpath); se

Condition →

```
if (!path.startsWith("/Auth")) {
```

Exception :-

Q. 1) Exception handle kese ki?

Ans → Spring ORM se.

It raise ORM specific exception by using DAO classes

It convert ORM specific exception to Data Access exception

Data Access exception is Unchecked exception

Q. 2) Data Access exception Unchecked exception kyu h?

(61)

Q. Query kha likhi he?

Ans → Base DAO me likhi he search ke liye.

Q. Project ko front end pr build kese kiya?

Ans → ng Build Se kiya the.

Q. Project ko back end pr build kese kiya?

Ans → Spring ORS Application men class me Spring Boot Initializer Se extend kiyo

Q. Checked exception kb aye gi?

Ans → compile

Q. Unchecked exception kb aye gi?

Ans → Run time pr.

Q. Service layer^{class} pr checked exception propagate hogi to kya hogi?

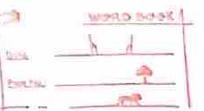
Ans → try catch lga for handle karte transaction commit hogi

Q. Service class pr Unchecked exception propagate hogi to kya hogi?

Ans → transaction Roll back liga exception propagate hogi

(62)

Base (1) (S+CH)



Q.1) next ke liye kom sg function call hua ha

Ans:- Next ke liye next call hoga

Next button bnayg or onclick pr next method call ki
next method "pageNo+1" ko thi or display call kiyg

Display Indirectly search call koggi

code:- this.form.pageNo+1 for display ko call kiyg

Q.2) FE se BE pr data kese dhya he?
HTTP client ke throw

BE se FE pr data

Ans Response class bnayi he uskg
Object Return kiyg he Rest controller me

Rest controller - JSON, it convert kryg

Q. DCP ko Configure kese kiyg?

Ans Spring Boot home by default hiccup config
properties add ki

(63)

transaction

DAO se exception raise hogi or propagate hoke service pr jayegi.

Service pr handle kate he to commit
ko dege

Service pr handle nahi hoti ya propagate
hoti he to transaction Roll back hogi

① Service pr unchecked exception ati he
Data access exception ati he.

Spring DAO has exception ko convert
karte he Data access exception me

o

(2)

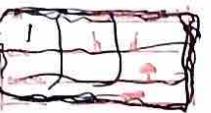
if

② Next pr click kiyg to kya hoga?
Search method call kiyg
Condition

③ previous pr code kya likha
Ans page No -- this.display

④ previous pr click kiyg to kya hoga?
services pr bki search method hi call kiyg

64



Q.1) Spring Boot me connection pool kese configuration
kijiye?

Ans → By default Spring ke andar RDBMS ke
Chikhi pool.

Q.2) Neat

Q.3) error ko kya handle kiyा?

Ans → Subscribe ki two methods he

- ① Success
- ② failer

error ayege to failer ke throw handle
kiye.

Q.

65

~~Base CTI~~ → login method bnayi he jo form
ka data legi

~~Base CTI~~ → ki validate method call hogi
some Binding Result ka object
pass kriye

Binding result . hasErrors true he

~~Success~~

HasErrors (true)
Success (false) kiyे he

~~Success~~

Q. Input msg kaise set kiyे

Ans → if (bindingResult.hasErrors()) {
 msg = "Success (false);
 HashMap ka object bnaya
 key value pair me. (e.g. get field)
 Def. addInputErrors(errors);
 return msg;
}

~~Final~~

~~Final~~

Angular (Imports: [RouterModule.forRoot(routes)]) 3)

const routes: Routes = [{ path: 'login', component: LoginComponent }]

Report frequent

@NgModule({
 imports: [RouterModule.forRoot(routes)],
 exports: [RouterModule]
})

Breadcrumb
Object - ke th
method ko .

AngularModule (Imports: [RouterModule.forRoot(routes)]),
Exports: [RouterModule]

<button (click) = "n
" btn btn-info" -

AngularModule

(Imports: [RouterModule.forRoot(routes)]),
Exports: [RouterModule]

<button (click) = "n
" btn btn-

AngularModule

(Imports: [RouterModule.forRoot(routes)]),
Exports: [RouterModule]

<button (click) := "n
" button button-

AngularModule

(Imports: [RouterModule.forRoot(routes)]),
Exports: [RouterModule]

<button (click) :="n
" button button-

AngularModule

(Imports: [RouterModule.forRoot(routes)]),
Exports: [RouterModule]

<button (click) :="n
" button button-

AngularModule

(Imports: [RouterModule.forRoot(routes)]),
Exports: [RouterModule]

<button (click) :="n
" button button-

AngularModule

(Imports: [RouterModule.forRoot(routes)]),
Exports: [RouterModule]

<button (click) :="n
" button button-

Angular

<button (click) :="n
" button button-

Click
Chay
148
47

[Engmodel] → form model ko import kiji
Import Books

Search field →

form eng! -

Jisme ~~text~~ input type text liji hoga

OSS [Engmodel] me form . Search param. first name. pass kiya

KO

- click eng! po submit method ko call kiya

- Component. ~~is~~ ^{mp} po submit method. bazi ne

OSS Usme Service locator. http Service

service locator. http service. Post([this](#).api. searching, this

form. Data)

dition

- Respective CT ^{Po} exted ^{kiya} baseCH ko baseCH

ki

Search method ko call ^{kiya}

ost. get

"v. " "j", ⁽³⁾ Base CH me OSS Response type ki gasey,
prefer brayi he. Jisme ⁽²⁾ Request body pass

ki he.

→ Public OSS Response search (Request body Form)

- OSS Response ki and result me key and value
pair me data ^{sa}

- OSS Response ka object brayi OSS Response
add path ko all kiya like under base
Service ki search ko call kiya or