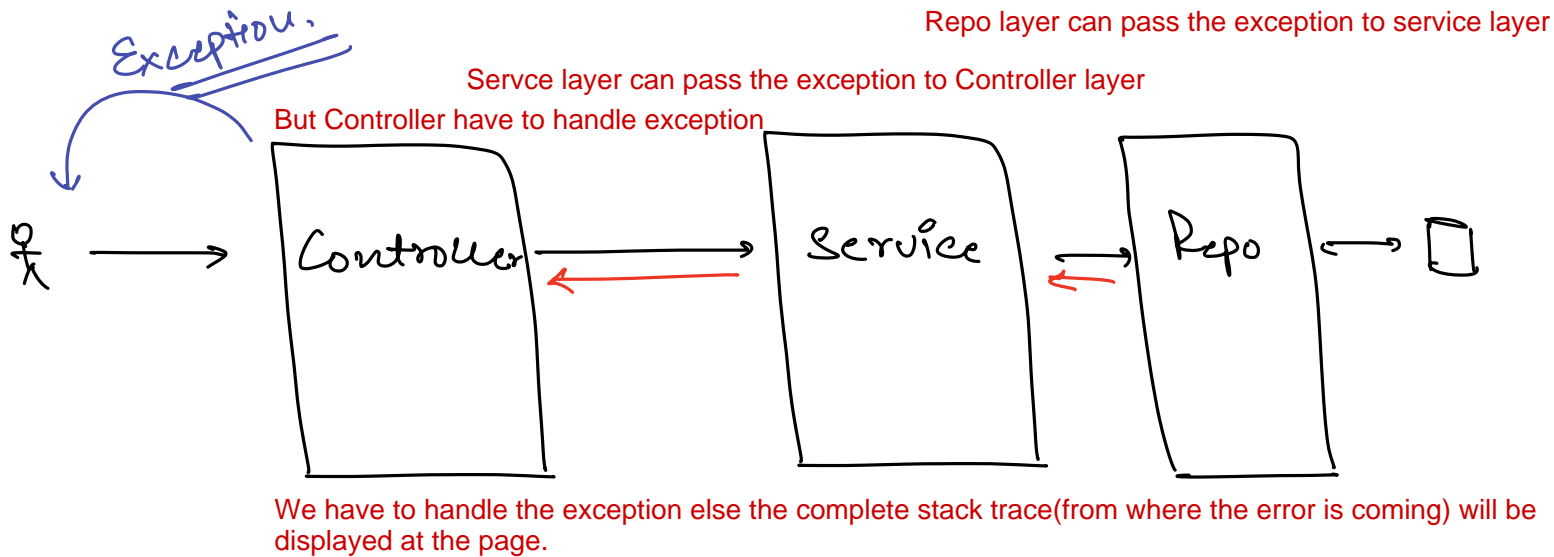


⇒ Exception Handling

⇒ Introduction to JPA.

⇒ Handle Exceptions in API's



fun() {

throw new RTE("Something went wrong");

}

ProductController {

add()

{ try {

fun();

} catch (—) {

3

3

We can handle exceptions for each catch block / specific errors.

catch(—) {

}

—

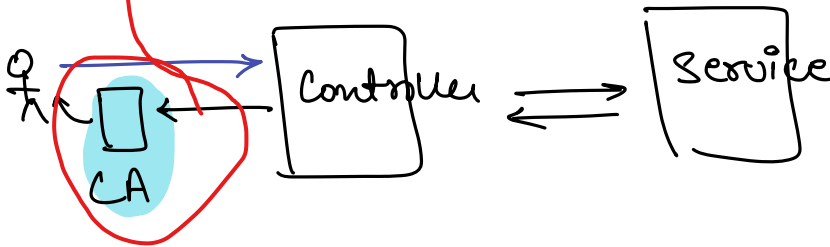
—

—

3

Ideally we should have exception specific catch blocks in order to do specific operation for each exception.

Controller Advice.



⇒ Controller is the final thing through which exception will be passed to the Client

Controller Advice ⇒ final check on whatever is being returned by the Controller

⇒ This final check can modify the data that is being returned to the Client.

Make a class for Controller Advice and annotate with @ControllerAdvice

This class would be said as Global Exception handler

Global Exception
↓
Handlers.

We mention controller advice here coz exception handler is for all controller is global so instead of writing too much code by writing try and catch block we use controller advice. Writing too much code in controller is not a good thing. We may want controller to be light

@ControllerAdvice

public class ExceptionHandler {

→ within this class there are going to be certain methods that can modify the content that is being sent to the client.

We define a method lets say named HandleArithmetic Exception. On top of this we will write @ExceptionHandler

@ExceptionHandler (ArithmeticException.class)

public void handleArithmeticExc() {

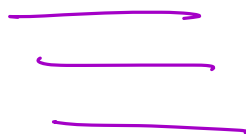
→ doSomething();



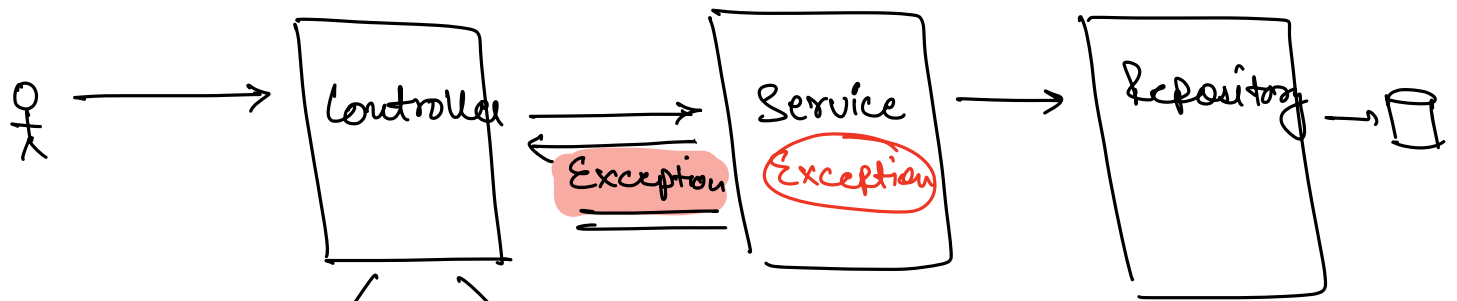
3
11

@ExceptionHandler (ArrayIndexOutOfBoundsException.class)

public void doSomething()



3



①

Handle
Exception



try Catch blocks
with in the
Controller.



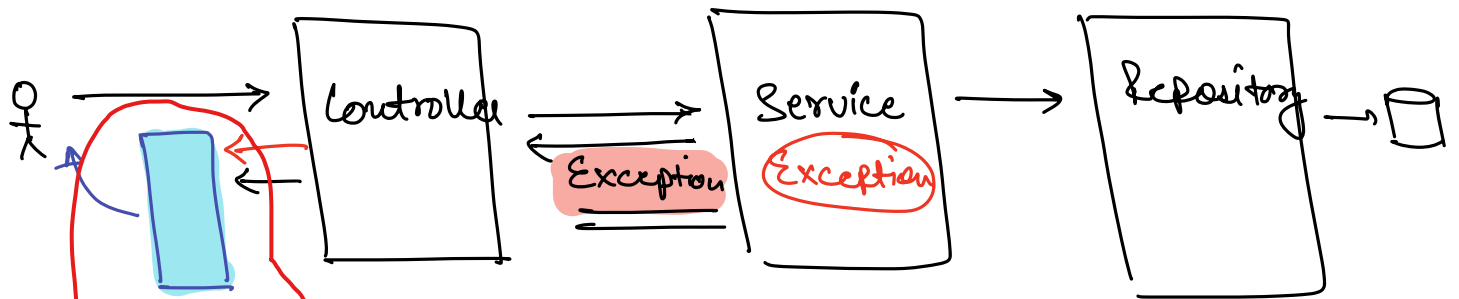
Not a good practice
of writing this much
logic inside the Controller

②

Pass the
Exception as it
is to the client.

Not Recommended.

Controller Advice



Controller Advice

Controller Advice is a global Exception Handler. So if we give annotation of @ControllerAdvice -> it will handle the exception instead of writing try and catch block

↳ Multiple Methods

@ExceptionHandler (———)

Throwable

Exception

Runtime Exception

NPE

Arithmetic Excep

SQL

FileNotFoundException

Unchecked

Checked