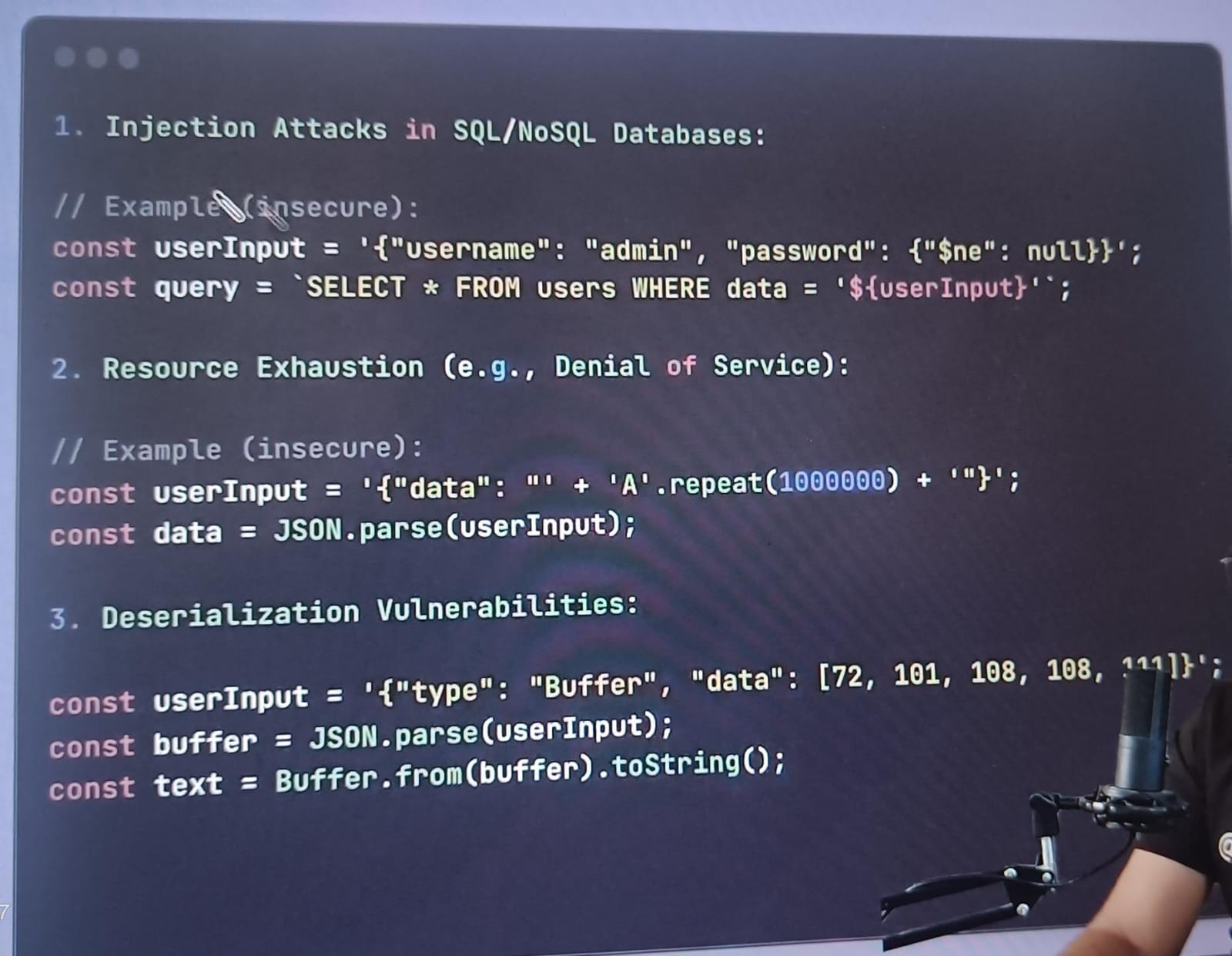
So, you have written a lot of JavaScript have we ever through the JS we are writing is secure enough? What if we write something on the server I Nodejs or express or any of these and we ended giving some access to the server, unwanted otherization people get some unwanted credential of your server which you never intended or probably there is some data breach basically happen which you never though about you JS code actually end up into.

* Inadequate Input Validation
* Direct execution of user provided code
* Using dangerous functions
* Insecure deserialization



Suppose this is a userInput in line no2, any how you got, you got from the user input field, from the form, form the url from anywhere basically you took this user data, now this is a user data which you expected to be a json format which is well and good or probably a string format which is well and good what you ended up doing? You ended up doing there was some database query that was written that you may be familiar with SQL injection also something similar to that was written and you ended up concatenating data which came directly form userInput(line no2).

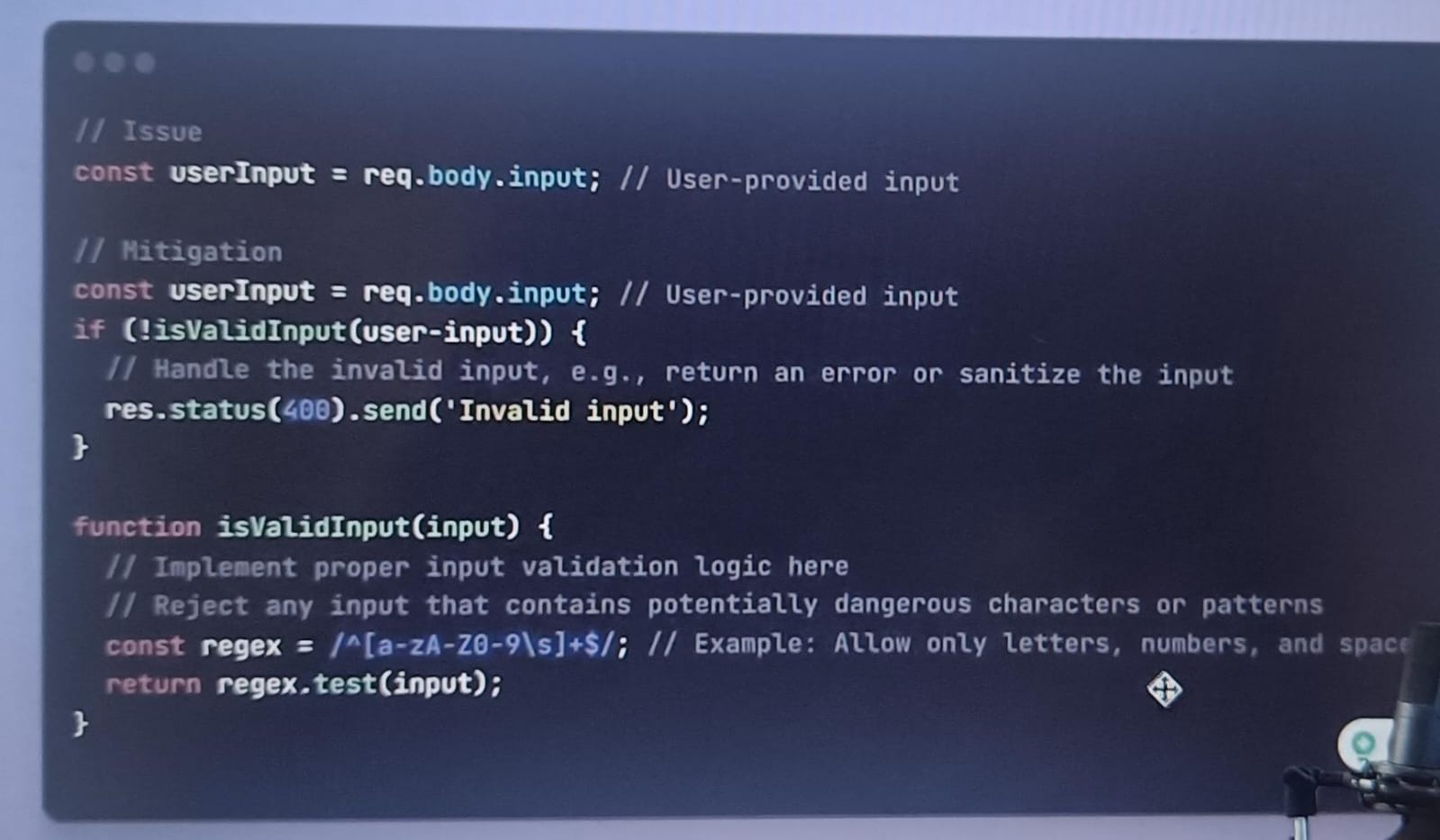
In 2nd code you got the user input where we are saying A.repeat() and you are doing parsing of this, when you try to parse it basically consume huge memory space, this kind of thing happen in the d dos attack also.

Now in 3rd code we talked about deserialization so if insure deserialization is handled something similar can happen, in this case the userInput is something called buffer kind of thing now you don’t know what kind of buffer data basically you are getting, what you just expected some data I will be getting in the userInput and you just did a parse and whatever you data get in the parse ad then you are trying to basically convert buffer into the text, now this can be a malicious stuff which can be executed which you don’t know.

Now how to take care of this –

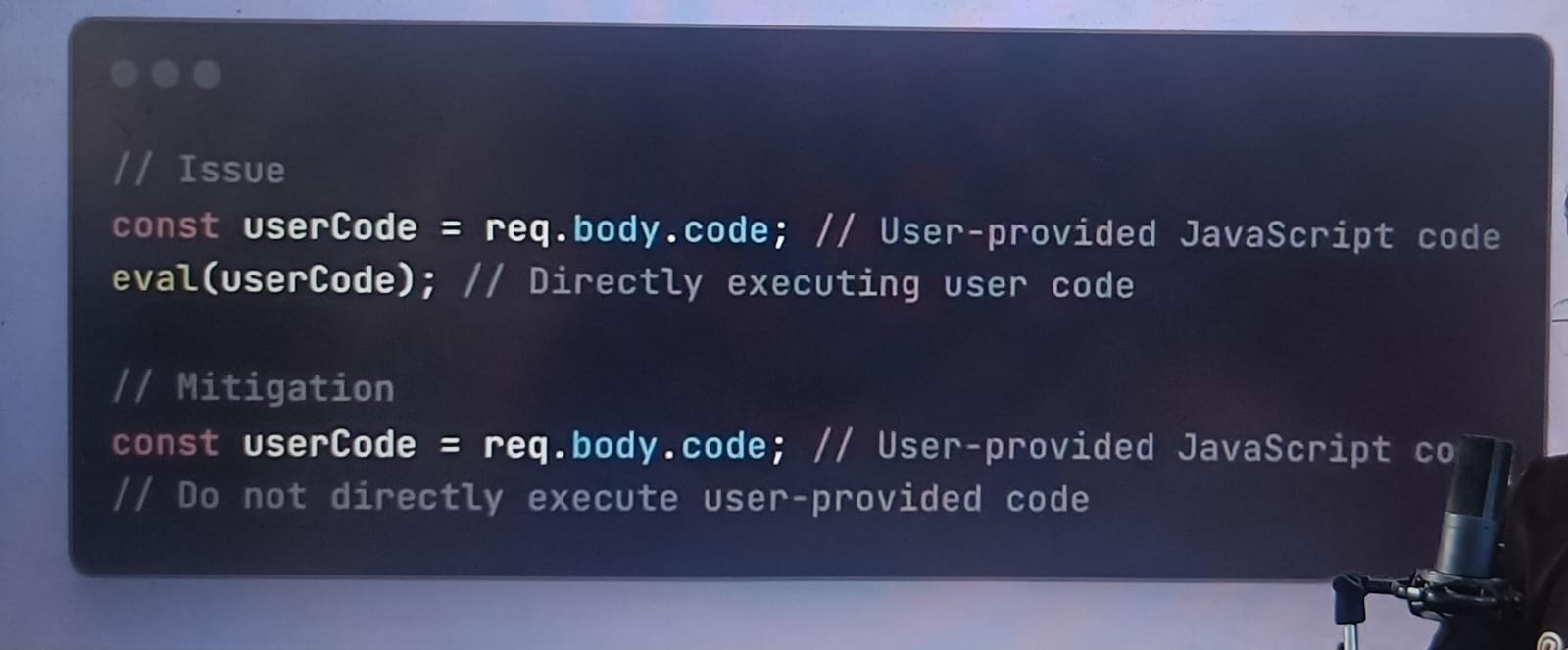
* Inadequate input validation –

In such case



You get the data from the body, so you have to always try to validate that data so there should be some method like regex, so you should allow only certain set of characters, letter or spaces whatever you fill is the right data for you which you should handle, you should not take dam any data which is coming from the user input, there should be some way to segregate that data, and if any data is not valid then please send invalid or a bad request.

* Direct execution of user provided code –

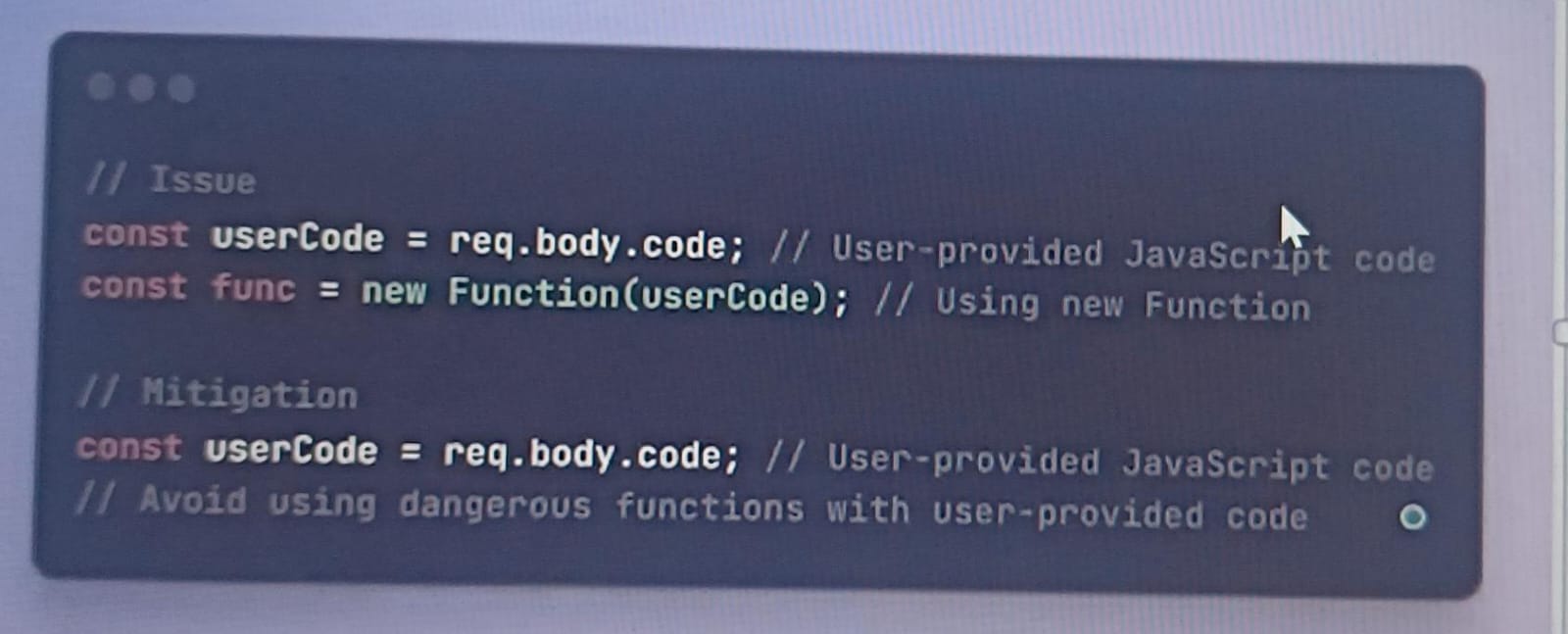


In this case we got some user data, it may be from body or any data and we blindly basically execute this code now this code can be any dam thing, now this code or this body can contain any dam string, like console.log() and you put that string into aval statement. So, it can any dam malicious code and if it came then it can execute it so what you should do basically is.

You should not use eval on your server, because if server is breached then you are going to be of nowhere right.

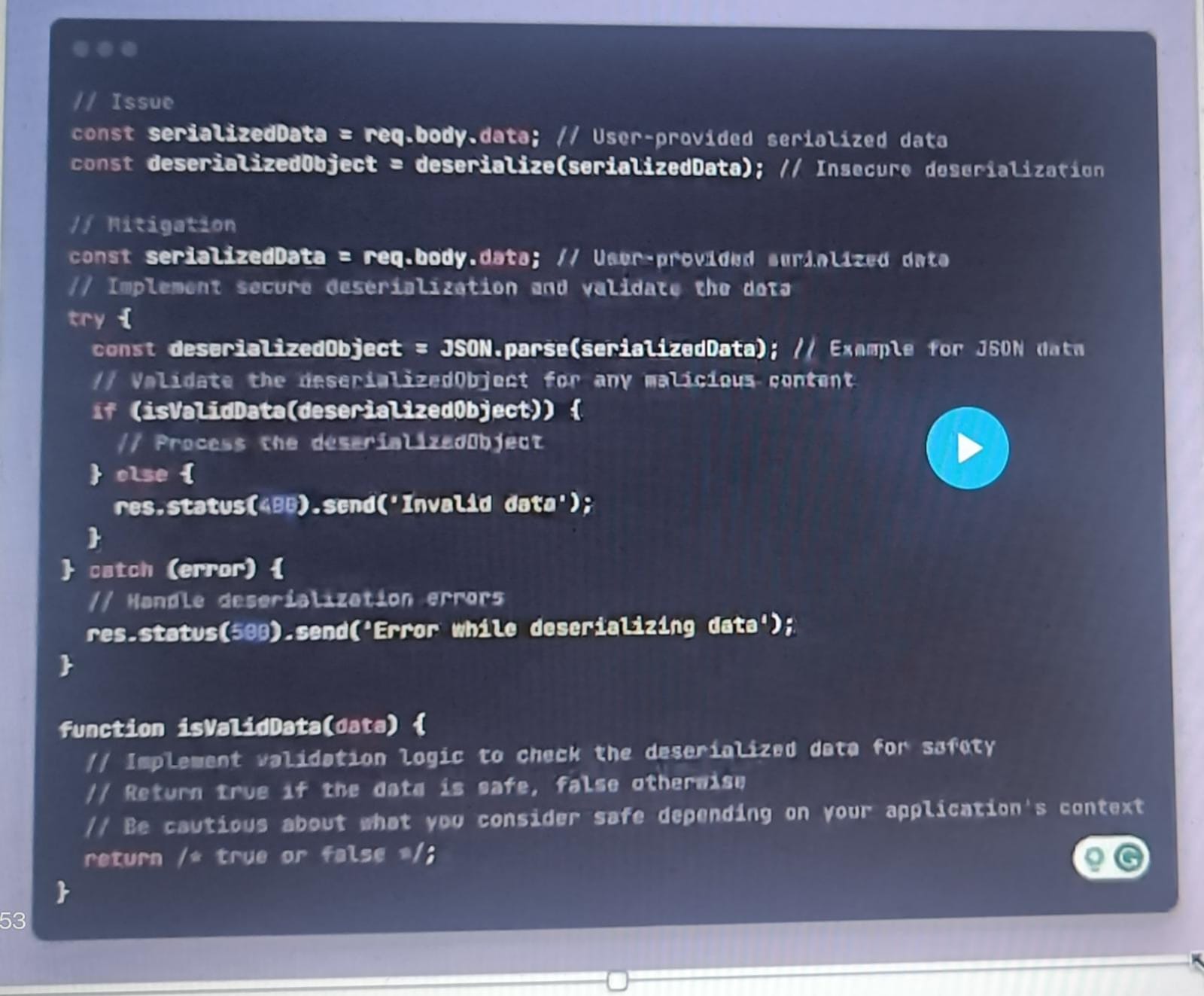
* Using some dangerous functions –

Example in this particular case is



So in this the code that you got from the req.body or a user you are using for blindly creating a new function out of that so this is something dangerous.

* Insure Deserialization –



So this is a insecure deserialization, in the first 2 line you get the data from the user and deserialize it blindly, but this is not correct because it can corrupt your system, what you should do?

You should basically first try to parse it , it there is a error in parsing then please do a proper exception handling for that and if this parsing is gone well then check if the data is valid or not, and if data is valid then its ok else basically you have to throw bad request.