Middlewares in Node.js, especially within the context of web development frameworks like Express.js, serve as functions that have access to the request object (req), the response object (res), and the next middleware function in the application’s request-response cycle. These functions can execute any code, make changes to the request and response objects, end the request-response cycle, and call the next middleware function in the stack.

Middlewares usually have next(), which will send the callstack to the next function.

What we use for exception handling-  
1) Status Code

2)Middlewares and Global Catches

When we are requesting a json from body, we do

const input=req.body;

and not req.body.input;

Whenever using a user-defined middleware, use (req,res,next)

next() in user-defined middlewares come at the end of the middleware function. They allow to let the program flow move on and go forward.