We will do 2 things in lecture-

1)We will see how we can extract the request body of incoming request

2)How we will handle CORS error and what CORS error are

Lets extract request body first. For this we w need to install a nw package-

**npm install –save body-parser**

we will use this to parse body because it is not nicely formatted and easily readable by default in nodejs. Body parser does not support files for example but it does support URL encoded bodies and it also support json data. Now wew ill use it-

first we import in in app.js then we apply it to our request. We use it as a middle ware after my logging middleware. Bodyparser needs additional information ,which kind of bodies we want to parse. First lets parse url-encoded bodies. then we pass a javascript object which then configures this kind of parsing. There we can set extended property to true or false. True allows you to parse extended bodies with rich data in it. We will set it to false to support simple bodies for URL encoded data.

Then we set up prser for json data. Now we can use body property on req in all our reqest handlers.

Now we use body property on req in post handler of /products(product.js file) route. Here we extract name property from body and use it.

Then we do same thing for post request handler for /orders.

Now save oyour code and send post request with body.

Now lets have look at CORS errors.

To prevent CORS errors we have to send some headers from server to browser. Server says- its ok you can have access.

Now go to app.js. now we want to append headers to any response that we sent back. So we should do this before request reaches our routes, so we add middleware before request reaches our routes.