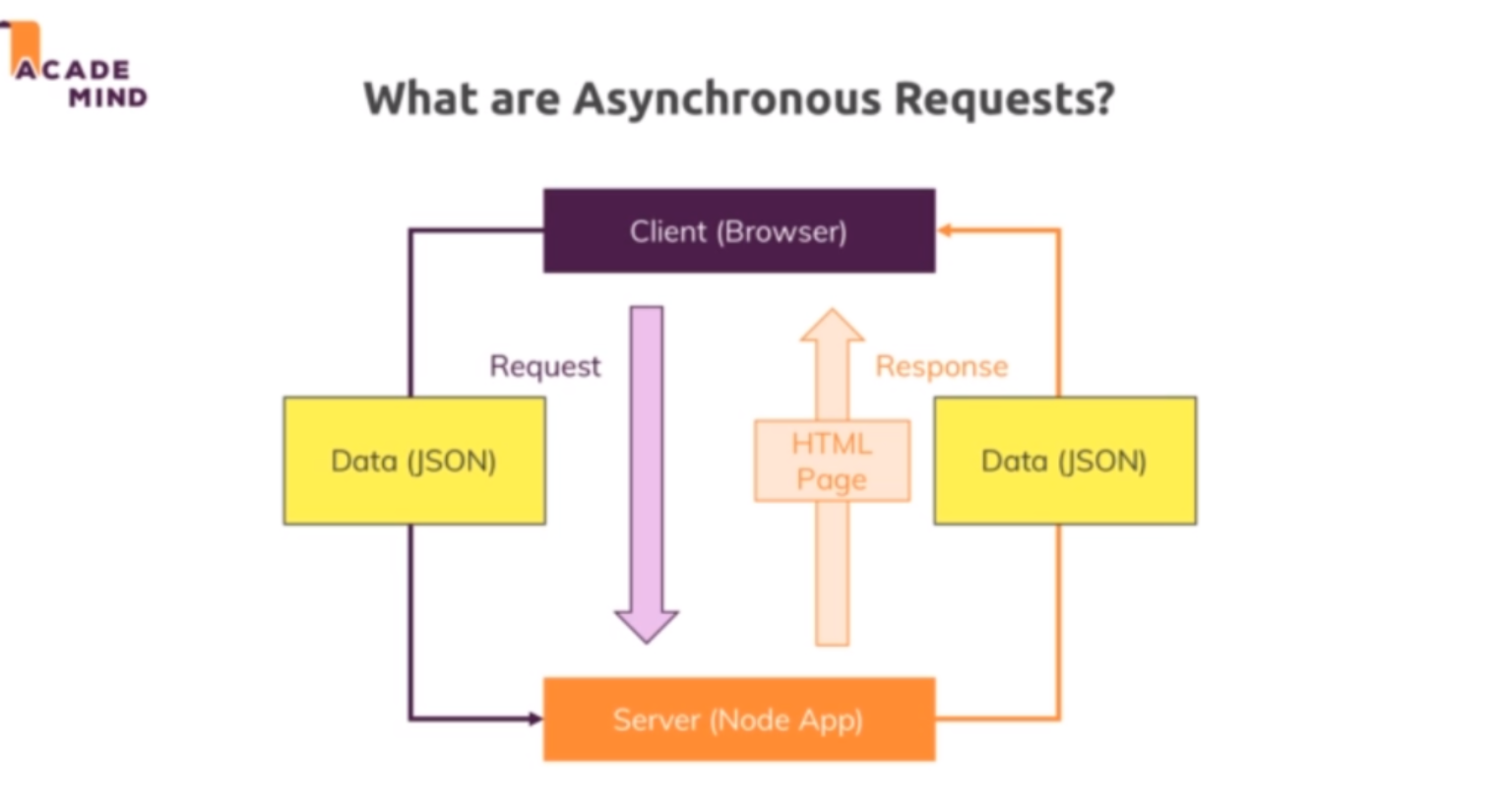
**Lecture 347**

**What are async requests**

* Refer notes
* 

**Lecture 348**

**Adding client side JS code**

* Refer code 01-adding-client-side-code
* Public/js/admin.js 🡪 new js file
* Views/admin/products.ejs 🡪 linked the admin.js file. Added the file to the bottom to ensure that the script will be executed after the entire dom has been parsed and rendered.

Removing the encapsulating form , changing type of delete button from submit to button

**Lecture 350,351**

**Sending and Handling background requests**

* Refer code 02-sending-and-handling-background-requests
* Routes/admin.js 🡪 new route /product/:productid
* Controlers/admin.js 🡪 added deleteProduct method
* We can modify the existing /delete-product POST route. But we are adding a new route /product/:productId DELETE
* Using form submission we can sent only get and posts
* 200 is the default status code for res.json()
* Well we can set headers and in the headers, we could encode our csrf token because we still need to attach this to our request and right now we are not doing that. We cannot send it in the request body because delete requests don't have a body but the good thing is the csurf package which we are using on a server does not just look into request bodies, it also looks into the query parameters and therefore we could add it there and it also looks into the headers. So there we can add a csrf token header, it will look for this key, you'll find all the keys for which it will look in the official doc
* My product element and you can name this constant however you want is basically my button and then there is a closest method provided by javascript and you pass a selector to closest which gives you the closest element with that selector and the closest ancestor element to be precise and there, I will simply use article because I only have one article in my ancestor history here for this button, so if I select my closest article, that should be the element I want to delete. So inside here let's say, I can call product element remove and that is a function that will not be supported in Internet Explorer, there you would have to access the parent node and then remove a child and that child would be the product element. So that is a code that works in every browser, remove would work in all modern browsers.