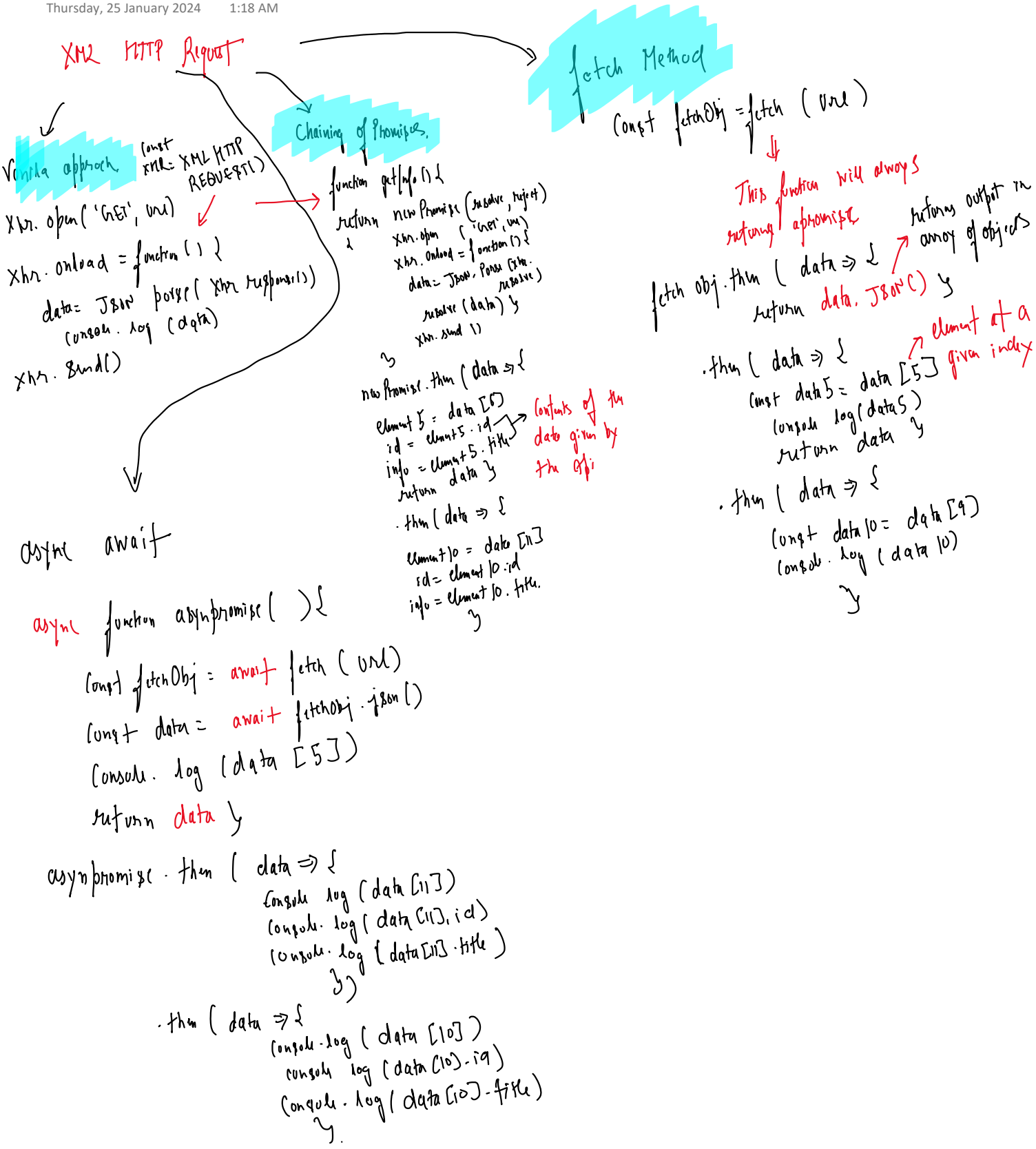
Day 14 Java Script

Thursday, 25 January 2024 1:18 AM




```
const xhr = new XMLHttpRequest() //creating xhr object
const url = 'https://jsonplaceholder.typicode.com/posts'
// xhr.open("GET", url)
```

```
// xhr.onload = function(){
//     console.log("Hello World of XML Server")
//     // console.log(xhr.response)
//     // getting the title of the third element
//     const realData = JSON.parse(xhr.response)
//     console.log(realData)
//     console.log("ID is ",realData[2].id)
//     console.log("Title is : ",realData[2].title)
// }
// xhr.send()
// // Working on getting the data with the process chaining of promises
// // ****************************************
function dataFromXhr() {
    return new Promise(function (resolve, revoke){
        xhr.open("GET", url)
        xhr.onload = function(){
            const data = xhr.response
            resolve(data)
            console.log(data)
        }
        xhr.send()
    })
}
dataFromXhr().then((data)=>{
    const parsedData = JSON.parse(data)
    const additionalPart = parsedData[3].id
    console.log(additionalPart)
    let ndata = `${url}/${additionalPart}`
    console.log(ndata)
    return ndata
    // returning a new url which is same url + id
}) .then(data =>{
    const url2 = data
    xhr.open('GET', url2)
    xhr.onload= function(){
        console.log("Meta data stored in the Data Base API: \n\n")
        console.log(xhr.response)
    }
    xhr.send()
})
// //  You can Even Simplify all above lines using Fetch method
const  promiseFetch = fetch(url)
promiseFetch.then(value=>{
    return value.json()
}).then(value=>{console.log(value)})
// You have simplified the above code in just 4 steps
const fetchObj = fetch(url)
fetchObj.then((data)=>data.json())
        .then((data)=>{ const data5 = data[5]
                        console.log('FETCHING INFORMATION :\n\n')
                        console.log("fetching data for id => ",data5.id)
                        console.log("Title extracted from the server for id 6
is :",data5.title)
                        return data})
        .then((data) =>{const data10 = data[9]
                        console.log("fetching data for id => ",data10.id)
                        console.log("Title extracted from the server for id 10
is :",data10.title)
                        return data})
// // Now using the async function along with await to get the similar resules
but in the more synchronous way
async function fetchAsyncfun(){
    try{
        console.log('Working with the Async and await functionality : ')
        const fetchObj = await fetch(url)
        if (!fetchObj.ok){
            console.log("Error Occured ")
        }
        else{
        const data = await fetchObj.json()
        // You can simply print information by this
        console.log("id is : ",data[2].id)
        console.log("Title is : ",data[2].title)
        console.log("Body is : ",data[2].body)
        return data
        }
    }catch{
        console.error('There is an Error')
    }
    // You can also chain this function since it returns a promise
```

```
}
fetchAsyncfun().then(value=>{
    const tenthElement = value[11]
    console.log(tenthElement)
    return tenthElement
}).then(value => {console.log('Title info from Id 10 :',value.title)})
```