React:

VS Code extensions needed: ES7+ React/Redux/React-Native snippets

App.jsx-> remove everything => for basic react template type rfce and enter you get the template

Component ek function hai jo ki return karega jsx

Jsx jo hai vo html ki tarah dikhne waala ek syntax hai par uske paas kuch superpowers hai jo html ke paas nahi hai.

HTML: <h1>2+2</h1> => output: 2+2

JSX: <h1>{2+2}</h1> => output: <h1>4</h1>

Jsx dikhta html ki tarah hai lakin baad mai convert hota hai js mai

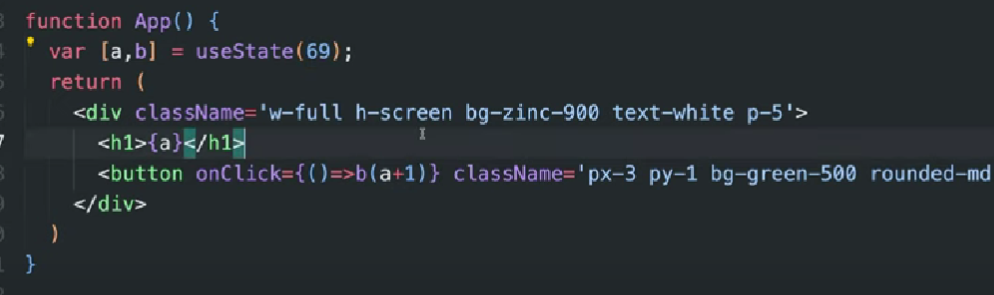
State: React mai jo value show hogi wo State se laai jati hai aur agar value change karni hai tab bhi value state mai change karenge jo phir jaake bahar reflect hogi.

Const [a,b] = useState(69) = => this means ki tum ‘a’ ke through 69 ko access kar sakte ho matlab agar kahi 69 print karna hai toh a print kar do and ‘b’ ke through tum uss value (69) ko update kar sakte ho.

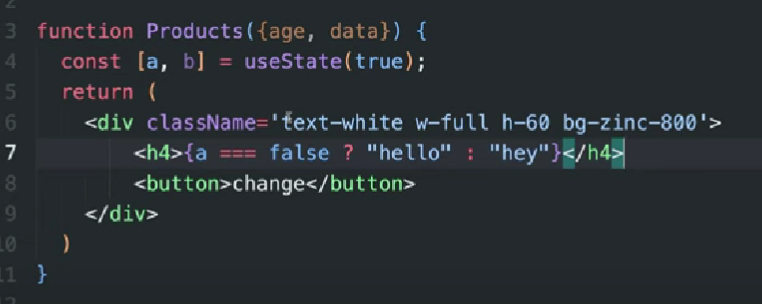
<h1>a</h1> => a

<h1>{a}</h1> => 69 <h1>{a+1}</h1> => 70

React ka naam react isiliye hai kyuki vo react karta hai jab bhi state change hoti hai aur yahi main model hai poore react ka.



Jab bhi button pe click karenge tab value 1 se badhti jaegi



**JS**

Hoisting: Variable ko banane se pehle use ksr sakte hai. Variables and functions are hoisted which means that their declaration is moved on the top of the code.

console log(a);

var a=12;

For the above 2 lines the output is undefined because JS converts the above 2 lines into:

var a;

console.log();

a=12;

Undefined: variable exists but its value is unknown.

Not defined: variable doesn’t exist.

Primitives & Reference:

Aisi koi bhi value jisko copy karne par real copy nahi hota, balki uss main value ka reference pass ho jata hai, use hum reference value kehte hai; aur jiska copy karne par real copy ho jae vo value primitive type value hoti hai.

Reference: [],(),{} : with bracket

Primitive: without bracket

var a = [1,2,3,4];

var b = a;

b.pop();

On printing b we get [1,2,3] and on printing a also we get [1,2,3]. var b =a ka matlab hota hai ki jo a ka hai vo hi b ka bhi hai. Agar b se vo khatam ho jaega toh vo a ke liye bhi khatam ho jaega.

function abcd(a,b,c){ // yaha pe a,b,c are parameters

Console.log(a,b,c);

}

abcd(1,2,3); yaha pe 1,2,3 are arguments

Arguments: real value jo hum dete hai function chakate waqt

Parameters: variables jinme value store hoti hai arguments wali

var arr = [1,2,3,4,5,6,7];

Arr.push(8); => [1,2,3,4,5,6,7,8]

var arr = [1,2,3,4,5,6,7];

Arr.pop(); => [1,2,3,4,5,6]

var arr = [1,2,3,4,5,6,7];

Arr.unshift(9); => [9,1,2,3,4,5,6,7]

var arr = [1,2,3,4,5,6,7];

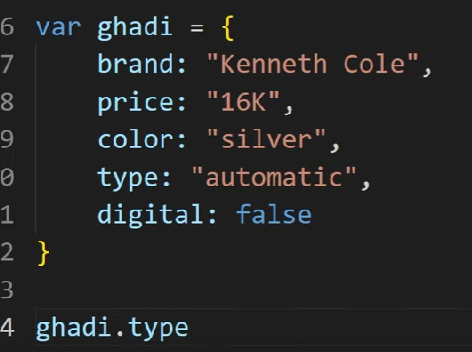
Arr.shift(); => [2,3,4,5,6,7]

var arr = [1,2,3,4,5,6,7];

Arr.splice(2,1); => [1,2,4,5,6,7] => 2nd index se 1 value hatao

Arr.splice(2,3); => [1,2,6,7] => 2nd index se 3 values hatao

Ek se zada bando ki baat ki toh hua array, aur agar ek bande ke baare mai saari baat ki toh hua object. Object hai ek bande ki details hold karna in key value pair.

making and accessing objects

Brand, price, color, type, digital are the properties

Ek aisi property jiski value ek function ho vo ek method hoga.

Express JS:

Bina express ke aap node js chala sakte ko lekin bina node js ke aap express js nahi chala sakte.

Ye ek package hai (npm pe available hai) jo server se related code likh kar deta hai aur http bhi same kaam karta hai lekin bas http mai code likhna bahut complex hai isiliye express introduce kiya gaya aur ab hum express use karte hai.

Node is the main thing, express ke code se hum server ka code likh paate hai and server kaise react karega vo bhi express ke help se hi likh paate hai.

Express ko routing ke liye use kiya jata hai.

Routes banane ke process ko routing kehte hai.

Routes => eg. /profile, /home, ya ‘/ ‘ ke saath kuch bhi jo url mai aata hai vo, /profile/palak/dxheiofmnesji



Middleware ek aisa function hota hai jo har route se pehle chalta hai, iska matlab saare routes mai koi bhi chale usse pehle middleware chalta hai and usme likha code pehle execute hota hai.

Middleware banane ka code:

app.use();-> ye use karna hai middleware ke liye. Iske andar ek function pass karna hai jisme 3 parameters pass karne hai ‘res,req,next’.

app.use(function(req,res,next){

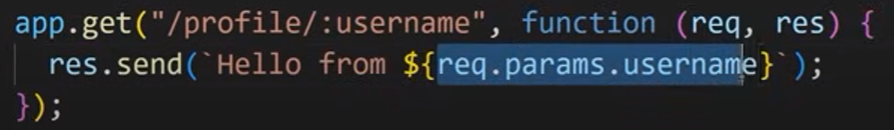
});

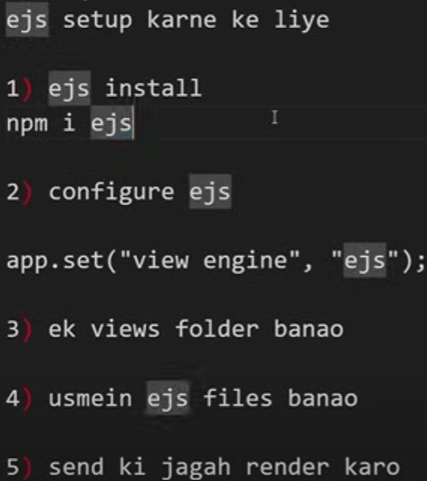
Req mai saara data hota hai aane wale user ki request ki taraf ka, jaise ki uski location, device info and other things, res mai controls hote hai jinke basis pe hum server se response bhej paate hai.

Req mai mera data hoga jo maine bheja/requwst kiya hai. Ab server jo response bhejega meri request ka vo res mai hoga. Req mai aane wala data, res mai saara yaha se jaane wala data.

Node ke saath ek dikkat hai ki agar control ek baar bhi kisi middleware par gaya toh control khudse agle route/middleware par nahi jaega, usse agle par le jaane ke liye aapko push karna padega aur ye push jo hai vo next se hota hai.

Aisa koi bhi route jiska koi hissa baar baar same reg=hta hai and kuch hissa baar baar change hota hai iske liye hum ek dynamic route bana sakte hai. To make any route dynamic we can use “:” at the place where we want to make it dynamic and to access their valuse use “req.params”.





Template engines: ye ek style of markup se convert karke aapko html dete hai. Egs of template engines: pug, jade, ejs, handlebars, etc. Ejs is very similar to html.

Internet, CN::::::::::::::

INTERNET:

Jab msg India ke kisi jagah se America bhejna ho:

Mujhe ek msg bhejna hai America mai kisi ko. Maine apne phone se ek message type kiya “HI” and send kiya. Toh ye msg packets ke form mai nearest tower tak pahuchaega aur vo tower phir uss packet ko electrical signals mai convert karega aur ptical fibres ke through usko ground and sea routes ke raaste America mai uss insaan ke sabse paas waale tower tak pahucha dega, jaha pe vo electric signals phir se packets mai convert honge aur vo packets uss insaan ke phone tak jaenge aur message usko receive ho jaega.

Lekin agar mujhe msg apne phone se apne bagal mai baithe hue aadmi ko bhejna hai tab:

Tower uss msg ko optical fibres ke through ISP ke paas bhejega waha ISP check karega ki request white listed hai ki black listed hai, agar white listed hai toh msg aage send hoga nahi toh agar black listed hai toh wahi pe block ho jaega.

ISP: Internet Service Provider: jo internet provide kar raha hai (jio, airtel, etc)

Client Server Architecture: Jo request bhejta hai vo clients hai aur jo uss request pe response bhejta hai vo server hai. The server need not have a screen, mouse, speakers, etc ; it is basically a CPU which receives the request and accordingly generates and sends the response.

http and https: inka kaam ek hi hai ki data ko internet pe bhejna aur aate hue data ko receive karwana. Bhejte aur aate waqt data ko safe and encrypted rakhna https ka kaam hai

http: Hyper Text Transfer Protocol

https: Hyper Text Transfer Protocol Secured

http ya https protocol ke bina tum internet pe na kuch bhej sakte ho aur na kuch manga sakte ho. Inhi 2 protocols ke through hume aaazadi milti hai ki hum internet pe surf kar sake.

http wala data koi bhi read kar sakta hai lekin https wala data encrypted hota hai aur koi bhi usko read nahi kar sakta.

Backend developer vo banda hai jo server and database program karta hai. Backend is used to make websites more useful and purpose full for the audience.

NODE JS:

Node js hai khoob saara c++ ka code jo ki liya gaya hai chrome browser ke v8 engine ke code se aur iss c++ code jisko liya gaya hai use wrap kar diya gaya js se and ye hume allow karta hai ki hum js code likhe aur ye js code accept karega and c++ ke basis par aapko ek server environment bana kar dega, par sirf tab tak jab tak ye code chalega jab tak aapka node (v8 engine and js wrapper) chalu hai.

Npm-> node package manager

Npm ek jagah hai jahan par khoob saare packages milte hai.

Bani banai cheezen/features ko packages kehte hai

Npm=>packages ka playstore | package = files of code

npx npm i nodemon -g

DATA ASSOCIATION:

Ek model se doosere model ke data ko jod dena, id ke through, matlab ki agar aapke paas ek user hai to wo post banaega hi, jab post banega to wo user ke dwara hi banega, toh hum kya karte hai jab do aise data aapas mai closely related hote hai toh hum dono ko jod dete hai, ki ek model ke bane hue data ki id ko doosere model ke data ko de dete hai aur iss model ke data ki id pichle model ke data ko de dete hai.