**Installation error- npm i -g** [**npm@8.5.1**](mailto:npm@8.5.1)

React.new –code sandbox

**Destructuring**

 it possible to unpack values from arrays, or properties from objects, into distinct variables.

Const fruit=[“App”,”ora”,’Pine’];

Let [p1,p2]=fruit

Console.log(p1,p2);

**Rest /Spread Operator**

Let Array=[1,2,3,4]

Let a2=[6,7,8]

Arr =[… Array,…a2] o/p=[1,2,3,4,6,7,8]

Template Literlas

It offers a simple method for performing string and interpolation multiline strng creation `${}`

If/else statements

Ternary operator

Ex:const pageno=pages >1000 ? “over a thousand”: “less than a thousand”

Used inside of template literals

Arrow function

Const no=(a,b,c)=>{

Console.log(a+b+c)}

No=(1,2,3)

Short circuit Logical operators(&& ,||)

Optional chaining

Const user={

Dog:{

Name:alex;

}

};

Console.log(user.dog?.name) //alex

Console.log(user.cat?.name) //undefined

Array map method

Const x=[1,2,3,4]

Const op=x.map((e)=>e\*2);

Console.log(op); [2,4,6,8]

Filter method

Const x=[1,2,3,4]

Const y=x.filter((e)=>e>2);

Console.log(y);[3,4]

Filter and map

Const x=[1,2,3,4]

Const y=x.filter((e)=>e>2).map((s)=>s\*2);

Console.log(y);[6,8]

Array reduce method

Reduce array to a single value executes and provide a function for each value and return value is stored in accumulator(acc).

Let val=[10,20,30,40];

Function sub(total,num){

Return total+num;

}

Console.log(val.reduce(sub);

Let val=[10,20,30,40];

Const arra=val.reduce((total,sum)=>total+sum);

Console.log(arra);

Array sort method

Const x=[1,2,3,4,5];

Const s=x.sort((a,b)=> b-a);

Console.log(s);[5,4,3,2,1]

Async Js function

Fetch(‘link’).then((res)=>res.json()).then(data=>console.log(data);

Async/await

Async function get data(){

Const res=await fetch (‘link’);

Const data=await res.json();

Console.log(data)

}

Getdata()

Rendering

With intermediate

Without intermediate

Conditional Rendering

{ function && ()}-short circuit

{ function ? (condition1):(cond2)}