CHAPTER 1: LEARNING JAVASCRIPT BASICS
(WEEK 1 DAY 1: 27 JULY 2023)

INTRODUCTION TO JS:

JavaScript is a programming language that allows you to implement complex functionalities on web pages. Every time a web page does more than just sit there and display static information for you to look at—displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, or more—you can bet that JavaScript is probably involved.

INTRODUCTION TO ES6:

ES6 or ECMAScript 2015 is the 6th version of the ECMAScript programming language. ECMAScript is the standardization of JavaScript which was released in 2015 and subsequently renamed as ECMAScript 2015. ECMAScript and JavaScript are both different in nature. ES6 introduced several key features like const, let, arrow functions, template literals, default parameters, and a lot more. Let's take a look at them one by one. "const" and "let"

OBJECTS:

In contrast, objects are used to store keyed collections of various data and more complex entities. In JavaScript, objects penetrate almost every aspect of the language. So we must understand them first before going in-depth anywhere else. An object can be created with figure brackets {…} with an optional list of properties. A property is a "key: value" pair, where key is a string (also called a "property name"), and value can be anything. We can imagine an object as a cabinet with signed files. Every piece of data is stored in its file by the key. It's easy to find a file by its name or add/remove a file.

Code:

```
<script>
                var objcases = {"Ahemedabad":[1500,1600,2],
"Rajkot":[500,600,1],"Vapi":[100,50,0]}
                for(var i in objcases){
document.write("<tr><td>"+i+"</td><td>"+objcases[i][0]+"</td><td
>"+objcases[i][1]+"</td><td>"+objcases[i][2]+"</td></tr>")
                }   </script>
```

```
//27 july file1 [object Object]
Ahemedabad
150
200
2200
```

(Fig 1.1)

(WEEK 1 DAY 2: 28 JULY 2023)

JS FUNCTION & ARROW FUNCTION:

There's another very simple and concise syntax for creating functions, that's often better than Function Expressions. It's called "arrow functions", because it looks like this:

```
let func = (arg1, arg2, ..., argN) => expression;
```

This creates a function func that accepts arguments arg1..argN, then evaluates the expression on the right side with their use and returns its result.

ASYNC FUNCTION:

The word "async" before a function means one simple thing: a function always returns a promise. Other values are wrapped in a resolved promise automatically. So, async ensures that the function returns a promise, and wraps non-promises in it. Simple enough, right? But not only that. There's another keyword, await, that works only inside async functions, and it's pretty cool.

BitCoin API:

Code:

```
<script>
          async function load(){
               let url
"https://api.coindesk.com/v1/bpi/currentprice.json"
               let coin =await(await fetch(url) ).json();
                    document.write(coin["bpi"]["USD"]["rate"])
               }
               load()
</script>
```

28,969.4077

(Fig 1.2)

CHAPTER 2: WORKING WITH API AND DATA MAPPING (WEEK 1 DAY 3: 31 JULY 2023)

COVID API:

Code:

```
<script>
async function getdata() {
let url = "https://data.covid19india.org/data.json"
let covin = await( await fetch(url) ).json();

document.write("<table><tr><th>Date</th><th>Confirmed</th><th>De
th</th><th>Recovered</th></tr>")
     for(var i in covin["cases_time_series"]){
          document.write("<tr> <td>
     "+covin["cases_time_series"][i]["date"]+"<td> ")
          document.write("<td>"+covin["cases_time_series"][i]["t
     otalconfirmed"])
     if(covin["cases_time_series"][i]["dailyconfirmed"]>0){
          document.write("<tag style='color:red'>
     (+"+covin["cases_time_series"][i]["dailyconfirmed"]+")</tag
     ></td>")
     } else{
     document.write("<tag style='color:green'>
(-"+covin["cases_time_series"][i]["dailyconfirmed"]+")</tag></td
>") }
document.write("<td>"+covin["cases_time_series"][i]["totaldeceas
ed"])
     if(covin["cases_time_series"][i]["dailydeceased"]>0){
document.write("<tag style='color:red'>
(+"+covin["cases_time_series"][i]["dailydeceased"]+")</tag></td>
")


     }  else{


document.write("<tag style='color:green'>
(-"+covin["cases_time_series"][i]["dailydeceased"]+")</tag></td>
")
                         }
```

```
     document.write("<td>"+covin["cases_time_series"][i]["totalr
ecovered"])
     if(covin["cases_time_series"][i]["dailyrecovered"]>0){
     document.write("<tag style='color:green'>
(+"+covin["cases_time_series"][i]["dailyrecovered"]+")</tag></td
>")
     } else{
     document.write("<tag style='color:red'>
(-"+covin["cases_time_series"][i]["dailyrecovered"]+")</tag></td
>")
                              } document.write("</tr>")
                         }  document.write("</table>")
                   }  getdata();
</script>
```

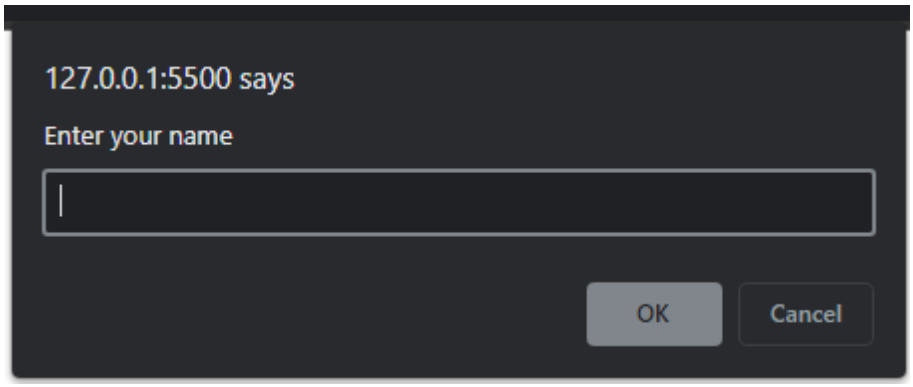
Date	Confirmed	Deth	Recovered
30 January 2020	1 (+1)	0 (-0)	0 (-0)
31 January 2020	1 (-0)	0 (-0)	0 (-0)
1 February 2020	1 (-0)	0 (-0)	0 (-0)
2 February 2020	2 (+1)	0 (-0)	0 (-0)
3 February 2020	3 (+1)	0 (-0)	0 (-0)
4 February 2020	3 (-0)	0 (-0)	0 (-0)
5 February 2020	3 (-0)	0 (-0)	0 (-0)
6 February 2020	3 (-0)	0 (-0)	0 (-0)
7 February 2020	3 (-0)	0 (-0)	0 (-0)
8 February 2020	3 (-0)	0 (-0)	0 (-0)
9 February 2020	3 (-0)	0 (-0)	0 (-0)
10 February 2020	3 (-0)	0 (-0)	0 (-0)
11 February	3 (-0)	0 (-0)	0 (-0)

(Fig-2.1)

USER INPUT USING PROMPT:

CODE:

```
<script>
          var name = prompt("Enter your name")
          document.write(name)
</script>
```

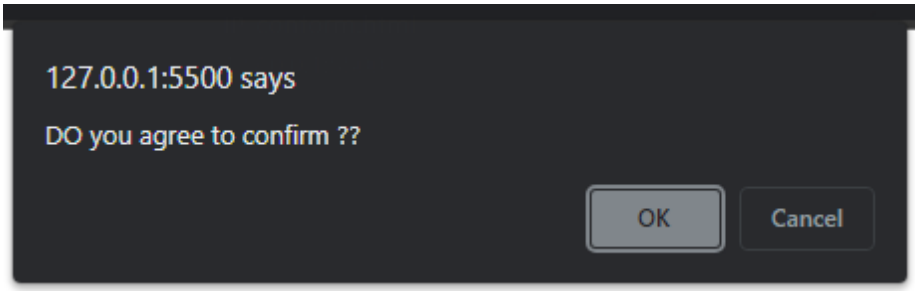


(Fig 2.2.1)

USER INPUT CONFIRM PROMPT :

CODE:

```
<script>
var con = confirm("DO you agree to confirm ??")
          if(con)
               document.write("you Agreed")
          else
               document.write("you dis-Agreed")
     </script>
```



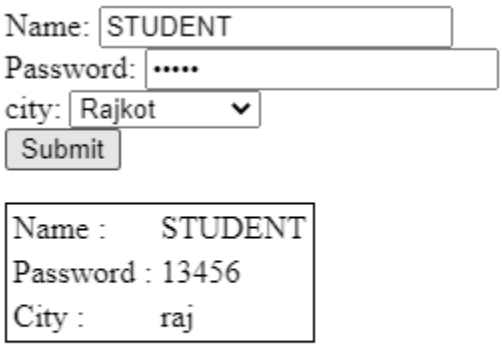
(Fig 2.2.2)

(WEEK 1 DAY 4: 01 AUGUST 2023)

FORM TO TABLE:

CODE:

```
<script>
          function showdata(form) {
               var name = form.uname.value
               var pas = form.upassword.value
               var city = form.ucity.value
               document.querySelector("#table").innerHTML =
"<table style=' border: 1px solid black;'> <tr> <td>Name : </td>
<td> "+name+"</td> </tr> <tr> <td> Password : </td> <td>
"+pas+"</td> </tr><tr> <td>City : </td> <td> "+city+"</td> </tr>
</table>";   }
     </script>
     <form class="form">
          Name: <input type="text" name="uname"><br>
          Password: <input type="password" name="upassword" >
<br>
          city: <select name="ucity" >
               <option value="ahm">Ahmedabad</option>
               <option value="raj">Rajkot</option>
               <option value="dhd">dhaod</option>
          </select><br>
          <input type="submit" onclick="showdata(this.form);
return false; " >
     </form>
```



(Fig 2.3)

JS ARRAYS:

Objects allow you to store keyed collections of values. That's fine. But quite often we find that we need an ordered collection, where we have a 1st, a 2nd, a 3rd element and so on. For example, we need that to store a list of something: users, goods, HTML elements etc. It is not convenient to use an object here, because it provides no methods to manage the order of elements. We can't insert a new property "between" the existing ones. Objects are just not meant for such use. There exists a special data structure named Array, to store ordered collections.

```
let fruits = ["Apple", "Orange", "Plum"];
```

CHAPTER 3: WEEKLY ASSIGNMENT-1
(WEEK 1 DAY 5: 02 AUGUST 2023)

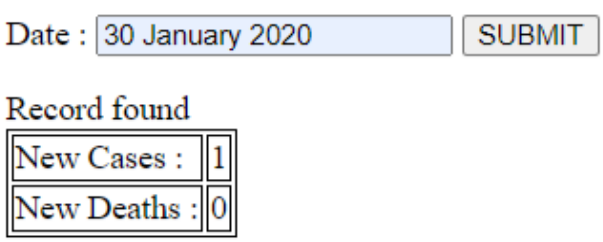
TASK ASSIGNMENT:

A weekly task based on the previously learned things throughout the entire week:

- Introduction to JS & ES6
- Arrays & Objects in JavaScript
- JavaScript Function, Arrow Function & Async Function
- Fetching API using JavaScript

COVID DATA SEARCH:

```
<script>
          async function CheckData(form) {
               var date = form.Datetof.value
               var result = document.getElementById("Result")
               result.innerHTML = " ";
               if(date==null){
                    result.innerHTML ="please enter valid Date "
                    return; }
               let URL =
"https://data.covid19india.org/data.json"
               let coinobj = await( await fetch(URL) ).json()
               for( var i in coinobj["cases_time_series"] ){
                    if( date ==
coinobj["cases_time_series"][i]["date"] )  {
     result.innerHTML += "Record  found <br> <table> <tr> <td>
New Cases :
</td><td>"+coinobj["cases_time_series"][i]["dailyconfirmed"]+"</
td></tr> <tr><td> New Deaths :
</td><td>"+coinobj["cases_time_series"][i]["dailydeceased"]+"</t
d></tr></table>" ;
                              return ; }
               } result.innerHTML =" not found "  }
</script>
```



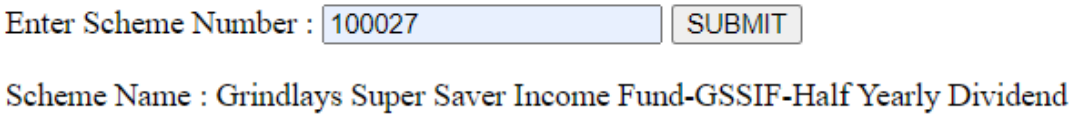
(Fig 3.1.1)

MUTUAL FUND API:

Code:

```
<script>
          function GetScheme(form) {
               var schemeCode = form.MFnumber.value
               if (schemeCode == "") {
                    result.innerHTML += "<br> Please Enter
Number :"
                    return;
               }
                    for( var i in MFobj){
                         if (schemeCode ==
MFobj[i]["schemeCode"]) {
                         result.innerHTML = "Scheme Name :
"+MFobj[i]["schemeName"]
                         return;
                         }
                    }
               result.innerHTML = "Scheme not found ::"
          }
     </script>
```

(Fig 3.1.2)



(Fig 3.1.3)

CHAPTER 4: LEARNING REACT BASICS
(WEEK 2 DAY 1: 03 AUGUST 2023)

REACT ENVIRONMENT SETUP:

Setting up a React environment involves several steps, including installing Node.js, creating a new React app, and configuring your development tools. Here's a basic guide to get you started:

Install Node.js and npm:

React apps require Node.js and npm (Node Package Manager) to manage dependencies and run scripts.
Download and install Node.js: Visit the official Node.js website (https://nodejs.org/) and download the LTS (Long Term Support) version for your operating system. Follow the installation instructions.

Create a New React App: Once Node.js is installed, you can use the create-react-app tool to quickly set up a new React application. Open your terminal and run the following command:

```
npx create-react-app react-app-name
```

Navigate to Your App Directory: Use the cd command in your terminal to navigate into the newly created app directory:

```
cd your-app-name
```

Start the Development Server: Inside the app directory, you can start the development server to see your app in action:

```
npm start
```

This will start the development server and automatically open your app in a browser. The server will also automatically reload the app whenever you make changes to the source code.

REACT FUNCTIONAL COMPONENTS:

A React functional component is a fundamental building block in React applications. It's a JavaScript function that returns JSX (JavaScript XML) to define the structure and behavior of a user interface component. Functional components are often used for creating UI elements that don't require internal state or lifecycle methods. With the introduction of React Hooks, functional components can also manage state and perform side effects, making them quite powerful.

```
function Welcome(props) {
     return <h1>Hello, {props.name}</h1>;  }
```

REACT CLASS COMPONENT:

A React class component is an alternative way to define components in React applications. While functional components have become the standard due to their simplicity and compatibility with React Hooks, class components were the primary way of creating components before the introduction of Hooks.

```
class Welcome extends React.Component {

  render() {

     return <h1>Hello, {this.props.name}</h1>;

  }

}
```

(WEEK 2 DAY 2: 04 AUGUST 2023)

VARIABLE DATA MAP:

CODE:

```
let aryone = [10, 11, 12, 13, 14, 15];
function AryMap() {
  return (
     <div>
     {
     aryone.map( (data, i) => {  return i + " = " + data + ", "
} )
     }
     </div>
  ); }
```

OBJECT MAP:

CODE:

```
let objone = {
  name: "Narayana",
  surname: "Murti",
}
function ObjMap() {
  return (
     <div>
     <h1>
     { Object.keys(objone).map(  (item) => { return objone[item]
+ "  "  } )
     }
     </h1>
     </div>
  );

}
```

API OBJECT DATA MAP:

CODE:

```
let apiISRO = { "spacecrafts": [{ "id": 1, "name": "Aryabhata" }, { "id": 2, "name": "Bhaskara-I" }, { "id": 3, "name": "Rohini Technology Payload (RTP)" }, { "id": 4, "name": "Rohini Satellite RS-1" }, { "id": 5, "name": "Rohini Satellite RS-D1" }, { "id": 6, "name": "APPLE" }, { "id": 7, "name": "Bhaskara-II" }, { "id": 8, "name": "INSAT-1A" }, { "id": 9, "name": "Rohini Satellite RS-D2" }, { "id": 10, "name": "INSAT-1B" }, { "id": 11, "name": "SROSS-1" }, { "id": 12, "name": "IRS-1A" }, { "id": 13, "name": "SROSS-2" }, { "id": 14, "name": "INSAT-1C" }, { "id": 15, "name": "INSAT-1D" }, { "id": 16, "name": "IRS-1B" }, { "id": 17, "name": "SROSS-C" }, { "id": 18, "name": "INSAT-2A" }, { "id": 19, "name": "INSAT-2B" }, { "id": 20, "name": "IRS-1E" }, { "id": 21, "name": "SROSS-C2" }, { "id": 22, "name": "IRS-P2" }, { "id": 23, "name": "INSAT-2C" }, { "id": 24, "name": "IRS-1C" }, { "id": 25, "name": "IRS-P3" }, { "id": 26, "name": "INSAT-2D" }, { "id": 27, "name": "IRS-1D" }, { "id": 28, "name": "INSAT-2E" }, { "id": 29, "name": "Oceansat(IRS-P4)" }, { "id": 30, "name": "INSAT-3B" }, { "id": 31, "name": "GSAT-1" }, { "id": 32, "name": "The Technology Experiment Satellite (TES)" }, { "id": 33, "name": "INSAT-3C" }, { "id": 34, "name": "KALPANA-1" }, { "id": 35, "name": "INSAT-3A" }, { "id": 36, "name": "GSAT-2" }, { "id": 37, "name": "INSAT-3E" }, { "id": 38, "name": "IRS-P6 / RESOURCESAT-1" }, { "id": 39, "name": "EDUSAT" }, { "id": 40, "name": "HAMSAT" }, { "id": 41, "name": "CARTOSAT-1" }, { "id": 42, "name": "INSAT-4A" }, { "id": 43, "name": "INSAT-4C" }, { "id": 44, "name": "SRE-1" }, { "id": 45, "name": "CARTOSAT-2" }, { "id": 46, "name": "INSAT-4B" }, { "id": 47, "name": "INSAT-4CR" }, { "id": 48, "name": "CARTOSAT - 2A" }, { "id": 49, "name": "IMS-1" }, { "id": 50, "name": "Chandrayaan-1" }, { "id": 51, "name": "RISAT-2" }, { "id": 52, "name": "Oceansat-2" }, { "id": 53, "name": "GSAT-4" }, { "id": 54, "name": "CARTOSAT-2B" }, { "id": 55, "name": "GSAT-5P" }, { "id": 56, "name": "YOUTHSAT" }, { "id": 57, "name": "RESOURCESAT-2" }, { "id": 58, "name": "GSAT-8" }, { "id": 59, "name": "GSAT-12" }, { "id": 60, "name": "Megha-Tropiques" }, { "id": 61, "name": "RISAT-1" }, { "id": 62, "name": "GSAT-10" }, { "id": 63, "name": "SARAL" }, { "id": 64, "name": "IRNSS-1A" }, { "id": 65, "name": "INSAT-3D" }, { "id": 66, "name": "GSAT-7" }, { "id":
```

```
67, "name": "Mars Orbiter Mission Spacecraft" }, { "id": 68,
"name": "GSAT-14" }, { "id": 69, "name": "IRNSS-1B" }, { "id":
70, "name": "IRNSS-1C" }, { "id": 71, "name": "GSAT-16" }, {
"id": 72, "name": "Crew module Atmospheric Re-entry Experiment
(CARE)" }, { "id": 73, "name": "IRNSS-1D" }, { "id": 74, "name":
"GSAT-6" }, { "id": 75, "name": "Astrosat" }, { "id": 76,
"name": "GSAT-15" }, { "id": 77, "name": "IRNSS-1E" }, { "id":
78, "name": "IRNSS-1F" }, { "id": 79, "name": "IRNSS-1G" }, {
"id": 80, "name": "CARTOSAT-2 Series Satellite" }, { "id": 81,
"name": "INSAT-3DR" }, { "id": 82, "name": "SCATSAT-1" }, {
"id": 83, "name": "GSAT-18" }, { "id": 84, "name":
"RESOURCESAT-2A" }, { "id": 85, "name": "INS-1B" }, { "id": 86,
"name": "INS-1A" }, { "id": 87, "name": "Cartosat -2 Series
Satellite" }, { "id": 88, "name": "GSAT-9" }, { "id": 89,
"name": "GSAT-19" }, { "id": 90, "name": "Cartosat-2 Series
Satellite" }, { "id": 91, "name": "GSAT-17" }, { "id": 92,
"name": "IRNSS-1H" }, { "id": 93, "name": "INS-1C" }, { "id":
94, "name": "Cartosat-2 Series Satellite" }, { "id": 95, "name":
"Microsat" }, { "id": 96, "name": "GSAT-6A" }, { "id": 97,
"name": "IRNSS-1I" }, { "id": 98, "name": "GSAT-29" }, { "id":
99, "name": "HysIS" }, { "id": 100, "name": "GSAT-11 Mission" },
{ "id": 101, "name": "GSAT-7A" }, { "id": 102, "name":
"Microsat-R" }, { "id": 103, "name": "GSAT-31" }, { "id": 104,
"name": "EMISAT" }, { "id": 105, "name": "RISAT-2B" }, { "id":
106, "name": "Chandrayaan2" }, { "id": 107, "name": "Cartosat-3"
}, { "id": 108, "name": "RISAT-2BR1" }, { "id": 109, "name":
"GSAT-30" }, { "id": 110, "name": "EOS-01" }, { "id": 111,
"name": "CMS-01" }, { "id": 112, "name": "EOS-03" }] }

function Display_ISROSpacecrafts() {
  return (
     <div>
     <h1> All ISRO spacecraft's name </h1>
     <table>
<tbody>  <tr key={'header'} >
          <th>ID</th>
          <th>Name</th>
          </tr>
          { Object.keys(apiISRO["spacecrafts"]).map(
               (item) => {
               return (   <tr key={item} >
```

```
                    <td>{apiISRO["spacecrafts"][item].id}</td>
                    <td>{apiISRO["spacecrafts"][item].name}</td>
                </tr>  );
                }
            )
            }
        </tbody>
</table>  </div>
  );
}
```

0 = 10, 1 = 11, 2 = 12, 3 = 13, 4 = 14, 5 = 15,

Narayana Murti

All ISRO spacecraft's name

ID	Name
1	Aryabhata
2	Bhaskara-I
3	Rohini Technology Payload (RTP)
4	Rohini Satellite RS-1
5	Rohini Satellite RS-D1
6	APPLE
7	Bhaskara-II
8	INSAT-1A
9	Rohini Satellite RS-D2
10	INSAT-1B
11	SROSS-1
12	IRS-1A
13	SROSS-2
14	INSAT-1C
15	INSAT-1D
16	IRS-1B
17	SROSS-C
18	INSAT-2A
19	INSAT-2B
20	IRS-1E
21	SROSS-C2
22	IRS-P2

(Fig 4.1)

CHAPTER 5: UNDERSTANDING REACT PROPS AND HOOKS (WEEK 2 DAY 3: 07 AUGUST 2023)

REACT PROP:

In React, "props" (short for "properties") are a way to pass data from a parent component to a child component. Props are a core concept that enables you to create dynamic and reusable components by configuring their behavior and content from the outside. Props are read-only, meaning that a child component cannot modify the props it receives directly.

```
const myElement = <Car brand="Ford" />;
function Car(props) {
  return <h2>I am a { props.brand }!</h2>;
}
```

REACT BOOTSTRAP:

React Bootstrap is a popular library that provides a set of pre-designed, responsive, and customizable UI components for building web applications with React. It's built on top of the Bootstrap framework, which makes it easy to create visually appealing and consistent user interfaces. Using React Bootstrap, you can incorporate components like buttons, forms, modals, navigation bars, carousels, and more into your React application without having to write all the HTML and CSS yourself.

The best way to consume React-Bootstrap is via the npm package which you can install with npm.

```
npm install react-bootstrap bootstrap
```

Importing Components You should import individual components like:

```
import Button from 'react-bootstrap/Button';
```

Because React-Bootstrap doesn't depend on a very precise version of Bootstrap, we don't ship with any included CSS. However, some stylesheets are required to use these components.

```
{
  /* The following line can be included in your src/index.js or
App.js file */
}
import 'bootstrap/dist/css/bootstrap.min.css';
```

(WEEK 2 DAY 4: 08 AUGUST 2023)

VARIABLE REACT HOOKS:

UseEffects:

useEffect is a crucial built-in React Hook that allows you to perform side effects in your functional components. Side effects include data fetching, subscriptions, or manually changing the DOM. The useEffect Hook is used to manage these side effects in a way that's consistent with React's component lifecycle.

The basic syntax of the useEffect Hook looks like this:

```
import React, { useEffect } from 'react';
function MyComponent() {
  useEffect(() => {


     return () => {


     };
}, [dependencyArray]);
```

UseState:

useState is a built-in React Hook that allows functional components to manage and update state. It provides a way to introduce local state to your components without needing to use class components and their this.state mechanism. State is a way to store and manage data that can change over time, such as user input, fetched data, or UI-related values.

Here's how you can use the useState Hook:

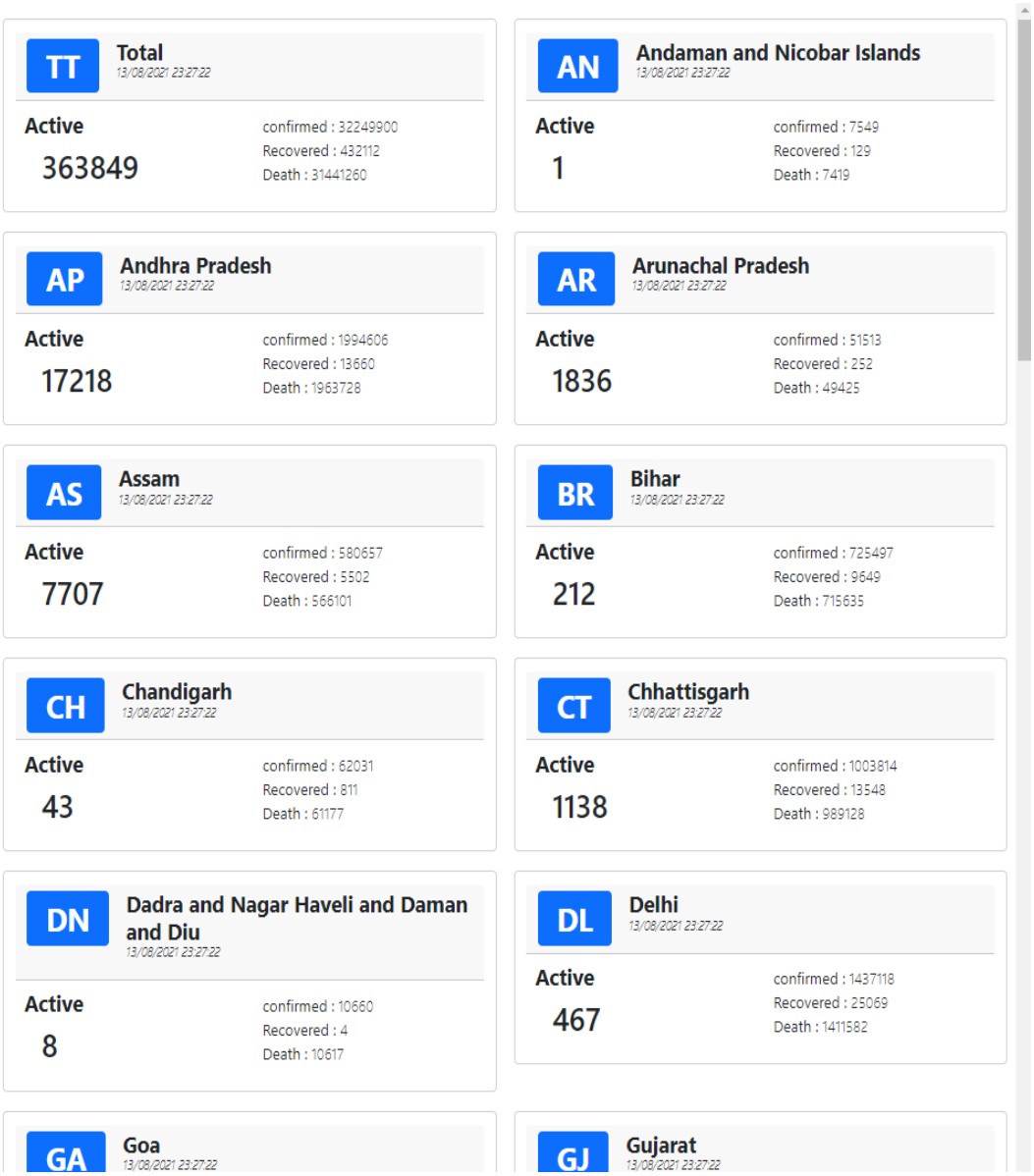
```
import React, { useState } from 'react';
function Counter() {
  const [count, setCount] = useState(0);
  const increment = () => {
     setCount(count + 1);  }
return (
     <div>
     <h2>Counter: {count}</h2>
     <button onClick={increment}>Increment</button>
     </div>
  );
}
export default Counter;
```

API DATA FETCH WITH REACT:

CODE:

```
import './App.css';
import Datacard from './DataCards';
import { Col, Container, Row } from 'react-bootstrap';
import { useEffect, useState } from 'react';
function App() {
  const [mydata,setData] = useState([]);
   const getapi = () =>{
     fetch('https://data.covid19india.org/data.json')
     .then((responce)=>responce.json())
     .then((json)=>{ setData(json.statewise) });
  }
  useEffect(()=>{
     getapi();
     const interval = setInterval(()=>{getapi();},5000000);
     return ()=> clearInterval(interval);
  })
   return (
    <Container fluid>
          <Row className='row-cols-2'  >
     {         mydata.map(  (values)=>{  return(
          <Col className='container-fluid mt-4'>

<Datacard data={values}  />
          </Col>  ) }  ) }
     </Row>
   </Container>
  );
}
export default App;
```


TT
Total
13/08/2021 23:27:22
Active
363849
confirmed : 32249900
Recovered : 432112
Death : 31441260
AN
Andaman and Nicobar Islands
13/08/2021 23:27:22
Active
1
confirmed : 7549
Recovered : 129
Death : 7419
AP
Andhra Pradesh
13/08/2021 23:27:22
Active
17218
confirmed : 1994606
Recovered : 13660
Death : 1963728
AR
Arunachal Pradesh
13/08/2021 23:27:22
Active
1836
confirmed : 51513
Recovered : 252
Death : 49425
AS
Assam
13/08/2021 23:27:22
Active
7707
confirmed : 580657
Recovered : 5502
Death : 566101
BR
Bihar
13/08/2021 23:27:22
Active
212
confirmed : 725497
Recovered : 9649
Death : 715635
CH
Chandigarh
13/08/2021 23:27:22
Active
43
confirmed : 62031
Recovered : 811
Death : 61177
CT
Chhattisgarh
13/08/2021 23:27:22
Active
1138
confirmed : 1003814
Recovered : 13548
Death : 989128
DN
Dadra and Nagar Haveli and Daman and Diu
13/08/2021 23:27:22
Active
8
confirmed : 10660
Recovered : 4
Death : 10617
DL
Delhi
13/08/2021 23:27:22
Active
467
confirmed : 1437118
Recovered : 25069
Death : 1411582
GA
Goa
13/08/2021 23:27:22
GJ
Gujarat
13/08/2021 23:27:22

(Fig 5.1)

CHAPTER 6: WEEKLY ASSIGNMENT-2
(WEEK 2 DAY 5: 09 AUGUST 2023)

TASK ASSIGNMENT:

Real time news fetching using news API. A weekly task based on the previously learned things throughout the entire week.

- *Functional Components & Class Components*
- *Object & Data Map*
- *React Props & Bootstrap*
- *React Hooks: UseEffect & UseState*
- *API data fetch in react using React Hooks*

News API :

App.js

```
  function upCatog(catg) {

     const [mydata, setData] = useState([]);

     useEffect(()=>{

fetch(`https://inshorts.me/news/topics/all?offset=0&limit=10`)
     .then((responce) => responce.json())
     .then((json)=> setData(json.data.articles) )
     .catch((err)=>console.log("err"+err));
     },[catg]);
  if (mydata.length >0 ) {
  return (
     <div className="main-title multi-column ">
     {mydata.map((values) => { return Cards(values); })}
   </div> );
   }else{
     return(

<h1>Data not found</h1>

);
  }
```

Netherlands slips into recession
It is Netherlands' 1st recession since pandemic
The Dutch economy has entered a recession as it shrank 0.3% on a quarterly basis in the second quarter, a first estimate published by Statistics Netherlands showed on Wednesday. The euro zone's fifth-largest economy shrank for the second consecutive quarter, after a 1.4% contraction in the first three months of the year.

Workers with traditional skills to get ₹1 lakh loan under Vishwakarma Scheme
PM Modi announced the scheme's launch yesterday
Union Railways Minister Ashwini Vaishnaw on Wednesday announced that PM Narendra Modi approved the Vishwakarma Scheme for workers having traditional skills like potters, blacksmiths, etc. The workers can get a loan of up to ₹1 lakh at a maximum interest rate of 5%, he added. Further, they will get support of up to ₹15,000 for purchasing modern tools.

₹6.9 cr spent for treatment of 3,400 'dead' patients under PMJAY scheme: CAG
The scheme was launched in 2018
The Comptroller and Auditor General of India (CAG) has said that ₹6.97 crore was paid for the treatment of 3,446 patients under the Ayushman Bharat – Pradhan Mantri Jan Arogya Yojana (PMJAY) health assurance scheme. These people were previously declared dead in the database. The most number of claims for 'dead' patients were paid in Kerala, the report added.
The Fed

(Fig 6.1)

CHAPTER 7: THE DAILY BUGLE
(WEEK 2 DAY 6: 10 AUGUST 2023)

PROJECT ASSIGNMENT :

A final internship project based on all the learning from internship, which consists of :

- Concepts of JavaScript
- Concepts of react
- Object data mapping and manipulation
- Handling API and its data
- Designing UI with Bootstrap
- NEWS WEB APPLICATION:

App.js

```
import './App.css';
import { Route, BrowserRouter as Router, Routes} from
'react-router-dom';
import Title, { Footer, One, Sidebar } from
'./components/Header';
import MainContent, { CatogaryNews } from
'./components/MainCon';
import { useEffect, useState } from "react";

 function App() {
   const [mydata, setData] = useState([]);
   let catogary = "all"

   useEffect(()=>{
     fetch("https://inshorts.me/news/topics")
     .then((responce) => responce.json())
     .then((json)=> setData(json.data.topics) )
     .catch((err)=>console.log("err"+err));
     },[]);



return (
     <main>
     <Title />

<One/>
```

```
<Router> <MainContent cat={catogary} adata={mydata}/>

<Routes>

<Route key="one" path='/' element={<CatogaryNews catg='all'
/>}/>
          {
          mydata.map( (values)=>{ return(
<Route key={values.hashId} path={"/"+values.topic}
element={<CatogaryNews catg={values.topic}/>} />

               );
               }
          )
          }

</Routes>


</Router>
     <Sidebar />
     <Footer />
     </main>
  );
}
export default App;
```

Components/Header.js

```
import { useEffect, useState } from "react";

function Title() {
  return (
     <h1> The Daily Bugle</h1>
     <aside>
     <div>
          <div  className="issue">{TodaysDate("day")}</div>
          <div className="date"> {TodaysDate("date")}</div>
```

```
          <div className="edition">new INDIA</div>
      </div>
      </aside>
      </>
    );
  }
function One() {
  const [mydata, setData] = useState([]);
  const apiget = () => {
      fetch("https://inshorts.me/news/trending?offset=0&limit=1")
      .then((response) => response.json())
      .then((json) => {
          setData(json.data.articles[0]);
      });
    };
    useEffect(() => {
      apiget();
      const interval = setInterval(() => { apiget(); }, 500000);
      return () => clearInterval(interval);
  }, []);


  return (
      <h2 className="title--large
main-title">{mydata.title}.</h2>
      <div className="main-text multi-column">
          <p>{mydata.content}</p>
          <p>{mydata.subtitle}</p>
          <p><em>Auther Name :</em> {mydata.authorName} <br />
<br /> <em>Source Name : </em>{mydata.sourceName}</p>
      </div>
      <a className="terrarium" href={mydata.sourceUrl}
target="_blank"  rel="noreferrer" >
      <figure>
          <img src={mydata.imageUrl} alt={mydata.imageUrl} />
          <figcaption>
          click above image to dicsover more ..
          </figcaption>
      </figure>
      </a>
  );
```

```
}
 function Footer() {
   return (
     <h1> The End. </h1>
   );
}
function Sidebar() {
  const [mydata, setData] = useState([]);
  const apiget = () => {
     fetch("https://inshorts.me/news/top?offset=0&limit=4")
     .then((response) => response.json())
     .then((json) => {
     setData(json.data.articles);
     });
  };


  useEffect(() => {
     apiget();
     const interval = setInterval(() => {
     apiget();
     }, 500000);
     return () => clearInterval(interval);   }, []);
   return (
     <div className="sidebar">
          <h3 className="title--big"> Today's Top </h3>
          {mydata.map((values) => {  return Sidecard(values);
})}
     </div>
      );
}

function Sidecard({hashId, subtitle, content, imageUrl,
sourceUrl, title }) {
return (
     <div key={hashId}>
     <a key={hashId} rel="noreferrer" style={{ marginBottom: 20
}} className="codepen-item pie"  href={sourceUrl}
target="_blank" >
     <img className="pie__image" src={imageUrl} alt={imageUrl}
/>
```

```
          <div className="pie__subtitle">{title} </div>
          <div className="pie__content">
          <h4>{subtitle}</h4>
          <p>{content}</p>
      </div>
      </a>
      </div>
  );
}

function TodaysDate(what) {
  const current = new Date();
  let day = current.getDay();
  const date = current.getDate();
  let month = current.getMonth();
  const year = current.getFullYear();

const dayname = [
"Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Sat
urday",
  ];

const monthname = [
"January","February","March","April","May","June","July","August
","September","October","November","December" ];

  day = dayname[day];
  month = monthname[month];
  const today = `${date} ${month} ${year}`;
  if (what === "day") return day;
  else if (what === "date") return today;
  else return day + today;

}

export default Title;

export { Footer, Sidebar, One };
```

Components/MainCon.js

```
 import { useEffect, useState } from "react";
 import { Link, Outlet } from "react-router-dom";
 function MainContent({cat,adata}) {
   return (
     <div className="catogary">
          <aside>
          <div>
          <div className="issue">
               <Link to="/" > <span id="catogaryName"> {cat}
</span> New's </Link>
          </div>
          <div className="issue">
               <Selector data={adata} />
          </div>
          </div>
     </aside>
     </div>
  );
}
function Selector({data}) {
   return (
     <div className="container">
  <button className="btn"><span>Change Catogary</span><i
className="material-icons"></i>
     <nav><ul className="dropdown">
     <li key={"all"}><Link to="/">All</Link></li>
     {
          data.map((values) => { return (
               <li key={values.topic}><Link
to={"/"+values.topic} >{values.label}</Link></li> );
          })
          }
     </ul></nav>
  </button>
</div>
<Outlet/>
  </>;
}
function CatogaryNews({ catg }) {
  const [mydata, setData] = useState([]);
```

```
  function upCatog(catg) {
     document.getElementById("catogaryName").innerHTML = catg; }
  useEffect(()=>{

fetch(`https://inshorts.me/news/topics/${catg}?offset=0&limit=10
`)
     .then((responce) => responce.json())
     .then((json)=> setData(json.data.articles) )
     .catch((err)=>console.log("err"+err));
     upCatog(catg);
     },[catg]);
  if (mydata.length >0 ) {
  return (
     <div className="main-title multi-column ">
     {mydata.map((values) => { return Cards(values); })}
     </div> );
  }else{ return(         <h1>Data not found</h1>     ); }
 }

function Cards({ hashId, sourceUrl, content, title, imageUrl })
{
   return ( <div className="cards" key={hashId}>
          <a className="item-with-image plan span--2 long--2 "
href={sourceUrl} target="_blank" rel="noreferrer" >
          <img alt={hashId} src={imageUrl} />
          <h4>{title}</h4>
          <div> <p>{content}</p> </div>
          </a>
     </div>
     );
   }
 export default MainContent;
 export {CatogaryNews};
```

OUTPUT:

THE DAILY BUGLE

WEDNESDAY | 16 AUGUST 2023 | NEW INDIA

Crocodile drags woman into river & eats her alive in Odisha; video surfaces.

A video has surfaced on social media that shows a crocodile dragging a woman into a river and eating her alive in Odisha. The incident took place in Jajpur when the woman was taking a bath near the river. As per reports, officials reached the spot and recovered the body of the woman from the river.

Locals were heard screaming in the video

Auther Name : Deepika Bharr

Source Name : YouTube

click above image to discover more..

ALL NEWS | CHANGE CATOGARY

Pune city reports a decline in COVID-19 testing numbers

The Pune Municipal Corporation (PMC) limits reported a decline in COVID-19 testing samples. The PMC collected 24,162 samples in August and 12,848 samples in July, which is less compared to previous months. Dr Ashish Bharti, chief, PMC health department, said, "Following a drop in COVID-19 cases and the festive season, many patients are not keen on getting tested."

BJP's Swapan Dasgupta, who lost WB polls, re-nominated to Rajya Sabha

Journalist Swapan Dasgupta, who had resigned from the Rajya Sabha in March to contest the West Bengal Assembly polls, has been re-nominated to the Upper House by President Ram Nath Kovind. Dasgupta was fielded by BJP from the Tarakeswar constituency and lost to TMC's Ramendu Sinharay. He had resigned from the Upper House on March 16, 2021.

Over 1,000 applications pending for SC/ST/OBC scholarships with Delhi govt

Over 1,000 applications for SC/ST/ OBC scholarships are pending with the Delhi government's education department by the end of 2021-22. The government's Department for the Welfare of SC/ST/OBC has written a letter to the Director of Education (DoE) requesting that all "pending applications are verified immediately".

Rahul will challenge PM Modi? PM is a gift of god to India: MP CM

MP CM Shivraj Singh Chouhan on Friday accused Rahul Gandhi of indulging in politics of poor taste, stating, "He'll challenge (PM Narendra) Modiji? Modiji is a gift of god to India." He added, "God no longer takes birth but whenever I see a leader like Narendra Modi, I feel no ordinary man can do so

Today's Top

WHAT ARE THE SCHEMES & PROJECTS APPROVED BY CENTRE TODAY?

Digital India will up-skill, train IT workers

The Union Cabinet on Wednesday approved several schemes and projects, including deployment of 10,000 electric buses in 169 cities for an estimated ₹57,613 crore. PM Vishwakarma Scheme for subsidised loans to traditional artisans and extension of the Digital India project were also approved. Seven Railways projects worth ₹32,500 crore to increase rail network by 2,339 km also got Cabinet nod.

NETHERLANDS SLIPS INTO RECESSION

It is Netherlands' 1st recession since pandemic

The Dutch economy has entered a recession as it shrank 0.3% on a quarterly basis in the second quarter, a first estimate published by Statistics Netherlands showed on Wednesday. The euro zone's fifth-largest economy shrank for the second consecutive quarter, after a 0.4% contraction in the first three months of the year.

WORKERS WITH

(Fig 7.1)

CHAPTER 8: CONCLUSION

In conclusion, the past two weeks of dynamic web development with React have been an incredibly enriching and transformative experience. Throughout this internship, I've delved into the world of modern web development, gaining hands-on experience with React and its ecosystem. From learning the fundamentals of component-based architecture to exploring state management, routing, and API integration, each day has brought new insights and challenges.

Over the course of these two weeks, I've had the opportunity to work on practical projects that have solidified my understanding of React's core concepts. The guidance and mentorship provided by the experienced development team have been invaluable in sharpening my skills and expanding my knowledge. Through collaborative coding, debugging sessions, and code reviews, I've not only improved my technical abilities but also gained exposure to industry best practices.

In closing, I am truly thankful for the invaluable experiences, lessons, and connections forged during these two weeks of dynamic web development with React. This internship has provided me with a solid foundation upon which I can continue to build as I pursue a career in web development.

CHAPTER 9: REFERENCES

https://developer.mozilla.org/en-US/

https://javascript.info/

https://react.dev/learn

https://react-bootstrap.netlify.app/docs/getting-started/introduction

https://data.covid19india.org/data.json

https://inshorts.me/news/topics/all?offset=0&limit=10

https://api.coindesk.com/v1/bpi/currentprice.json