Public index.html

<!DOCTYPE html>

<html lang="en">

  <head>

       <title>React App!</title>

  </head>

  <body>

    <div id="root"></div>

      </body>

</html>

Src Index.js

import React from "react";

import ReactDOM from "react-dom";

import App from "./App";

ReactDOM.render(<App />, document.getElementById("root"));

Src App.js

import React from "react";

import logo from "./logo.svg";

import "./App.css";

import { Comment } from "./MyComment";

function App() { //function component

  return (

    <div className="App">

      <header className="App-header">

        <img src={logo} className="App-logo" alt="logo" />

        <Comment zohiogch="Номин" />

      </header>

    </div> );}

export default App;

Src App.css

.App {

  text-align: center;

}

.MyCoolButton {

  width: 250px;

  height: 40px;

  font-size: 18px;

}

@media (prefers-reduced-motion: no-preference) {

  .App-logo {

    animation: App-logo-spin infinite 20s linear;

  }

}

.App-header {

  background-color: #282c34;

  min-height: 100vh;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  font-size: calc(10px + 2vmin);

  color: white;

}

.App-link {

  color: #61dafb;

}

@keyframes App-logo-spin {

  from {

    transform: rotate(0deg);

  }

  to {

    transform: rotate(360deg);

  }

}

Src Comment.js

import React, { Component } from "react";

import { MyCoolButton } from "./MyCoolButton";

export class Comment extends Component { // Class component

  constructor() {

    super();

    this.state = {

      publisher: "",

      comment: ""

    };

  }

  bichih = e => {

    this.setState({ comment: e.target.value });

  };

  render() {

    return (

      <div>

        <div>{this.state.comment}</div>

        <div>Зохиогч: {this.props.zohiogch}</div>

        <textarea onChange={this.bichih}></textarea>

        <MyCoolButton

          comment={this.state.comment}

          tovchNer="Комментоо үнэнээр нь бичээрэй.."

          text="Хадгал"

        />

      </div>

);

}

}

Src Button.js

import React from "react";

export const MyCoolButton = props => { // function arrow component

  const clickHandler = () => alert("Коммент: " + props.comment);

  return (

    <div>

      <span>{props.tovchNer ? props.tovchNer : "Гоё товч"}</span><br/>

      <input

        className="MyCoolButton"

        type="button"

        onClick={clickHandler}

        value={props.text ? props.text : "Click Me!!!!"}

      />

    </div>

  );

};

**Тайлбар**

React –> UI харуулдаг үндсэн react library бүх юманд ашиглана.

React-dom --> web app хийхэд ашигладаг.

React-scripts --> build хийх script-тай ажиллах гэх мэт ашигладаг.

JSX - javascript-ын өргөтгөл

**FUNCTION COMPONENT:**

Анх react-д class component хадгалж дамжуулдаг байсан. Function component persention маягын үзүүлэнгийн маягаар ажиглаж байгаад class component томрох тусам төлөвөө удирдахад төвөгтэй болж эхэлсэн. Тэгээд function component-д төлөв хадгалж hooks ашиглаж нэг газараас state –ээ удирдаж алдаа гарах магдлал бага болсон.Class component өөрөө төлөвөө хадгалдаг.

function component бол дотороо төлөвөө хадгалж дамжуулахдаа hooks ашигладаг.

function App() {

  return (

    <div className="App">

      <header className="App-header">

        <img src={logo} className="App-logo" alt="logo" />

        <Comment zohiogch="Номин" />

      </header>

    </div>

  );

}

export default App;

Нэгээс олон юм div бичих гэж байгаа бол гадуураа нэг хүрээлсэн div байх ёстой jsx нэг дүрэм. Return () хийж өгөхгүй бол return; гээд доод талын мөрөнд байгаа div буцахгүй алдаа заадаг байгаа.

1. const Mybutton = () => <input type="button" value="click me" />;

const Mybutton = () => React.createElement(‘input’, ……);

react package цаанаа эхлээд энэ файлыг compile хийгээд javascript болгочихдог.

Arrow function ашиглаж ганц элемент буцаах гэж байгаа бол ингэж бичиж болдог.

1. const Mybutton = () => (

<div>

<h2>nice</h2>

<input type="button" value="click me" />;

</div>

); Хэрвээ ингэж хийх тохиолдолд суман функцын () хаалт өөрийгөө буцаадаг байгаа.

**CLASS COMPONENT:** Component нь react package-ын component-ээс удамшдаг.

import React, { Component } from "react";

import { MyCoolButton } from "./MyCoolButton";

export class Comment extends Component {

  constructor() {

    super();

    this.state = {

      publisher: "",

      comment: ""

    };

  }

  bichih = e => {

    this.setState({ comment: e.target.value });

  };

  render() {

    return (

      <div>

        <div>{this.state.comment}</div>

        <div>Зохиогч: {this.props.zohiogch}</div>

        <textarea onChange={this.bichih}></textarea>

        <MyCoolButton

          comment={this.state.comment}

          tovchNer="Комментоо үнэнээр нь бичээрэй.."

          text="Хадгал"

        />

      </div>   );  } }

Named component import хийж оруулж ирэхдээ { } хаалтан дунд хийж өгдөг.

- constructor(){ Байгуулагч функц нь: component class-ын байгууллагч функцыг дуудаж өгдөг.

- super() Дээд талын байгууллагч цункцыг дуудаж өгнө. Ингэснээр төлөвтэй ажиллах боломжтой болдог. }

- this.state = {

words: []Component болгон state буюу төлөвтэй тэр дотороо хадгалдаг. Тэр төлөв дотор state гэдэг нэртэй төлөв дотор words гэсэн нэртэй массив байгаа гээд үзий. }

state нь component дотор байдаг төлөв хадгалах функц юм state өөрчилөхдөө заавалчгүй setState гэсэн өөрчилдөг функцаар өөрчилж өгдөг. Мөн адил component дотор байдаг функц удамшаад ороод ирдэг.

- render(){ Class component дотороо render функцтай байх ёстой. Дараах jsx буцааж байх ёстой

- return <div>welcome to</div> доторх код бол JSX код юм. HTML ижилхэн харагддаг. цаанаа javascript-руу хөрвүүлдэг байгаа }

Input дээр юм бичигдэхэд ямар event ажиллах вэ гэхээр onChange={this.handler}өөрийн функцруу хадгалах. Handler гэдэг өөрийн функц юм.

onChange дээр бичигдсэн зүйлийг авахдаа javascript-ын энгийн онол event.target.value гэж авдаг

this.state = {

      comment: ""

    };

  }

  handler = e => {

    this.setState({ comment: e.target.value });

  };

setState нь state-ын comment гэсэн объектруу input дээр бичигдсэн зүйлсийг merge хийж байна.

**CLASS PROPS**

App.js component

<Comment zohiogch="Номин" />

Comment.js component

<div>Зохиогч: {this.props.zohiogch}</div>

**FUNCTION PROPS**

Props javascript-ын объект юм.

Ингээд дамжуулчихаж болно. Гуравласан оператор ашиглаад нөхцөл шалгаж болно. ? --> тийм бол : үгүй бол

import React from "react";

import "./styles.css";

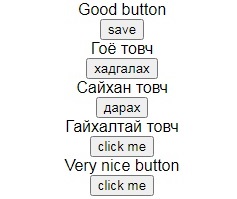
const Button = (props) => {

const clickHandler = () => alert(`Өнөөөдөрийн : ${new Date()}`);

return (

<div>

<span>{props.buttonName ? props.buttonName : "Very nice button"}</span> <br />



<input onClick={clickHandler}

type="button"

value={props.text ? props.text : "click me"} />

</div> );};

function App() {

return (

<div className="App">

<Button buttonName="Good button" text="save" />

<Button buttonName="Гоё товч" text="хадгалах" />

<Button buttonName="Сайхан товч" text="дарах" />

<Button buttonName="Гайхалтай товч" />

<Button />

</div> );}

export default App;

event-д хариу үзүүлдэг функцуудыг нэг бол Listenner нэг бол Handler гэсэн үгийг ашиглаж developer-ууд бичдэг байгаа.

Onclick гэдэг функц байдаг button дээр дарахад ажилладаг event юм.

HTML дээр onclick=”clickHandler()”

React дээр onClick={clickHandler} ингэж бичиж ажиллуулна. onclick хийгдэх үед л ажиллана.

React дээр onClick={clickHandler()} газар дээрээ ажилчихдаг байгаа.

const clickHandler = () => alert(`Өнөөөдөрийн : ${new Date()}`);

<input onClick={clickHandler}

type="button"

value={props.text ? props.text : "click me"} />

button товч дарангуут өнөөдрийн он сар өдөр харуулж байна.

Templete string дотор javascript ашиглах гэж байгаа бол ${new Date()} бичдэг.

Button.js

const clickHandler = () => alert("Коммент: " + props.comment);

onClick={clickHandler}

Суман функц бичээд дотор нь байгаа handler руу аргемент хэрхэн дамжуулах вэ гэдгийг харуулж байна.

const clickHandler = (c) => alert("Коммент: " + c);

onClick={() => clickHandler(props.comment)}

**СЭРВЭРЭЭС ӨГӨГДӨЛ JSON БАЙДЛААР ТАТАЖ АВЧИРААД ХАДГАЛАХДАА**

Class component дээр амьдралын цикл гэж байдаг. Хамгийн анх render хийж гарч ирэх юмуудыг тодорхой дараалаар зарим функцууд дуудагддаг байгаа.

Render гэдэг функц дуудагдаад гарч ирсэний дараа сэрвэрээс татах үйлдэлээ хийх хэрэгтэй.

import React { Component } from "react";

export default class App extends Component {

constructor () {  
super()

this.state = {

robots: []

}

componentDidMount(){

fetch(‘https://jsonplaceholder.typicode.com/users’)

.then(response => response.json())

.then(data => this.setState({ robots: data }));  
}

render() {

console.log(this.state.robots)

return <div>Robots</div>

}; };

Render-ын дараа ажилладаг react –ын тусгай class component –ын функц бол componentDidMount(){}

fetch(‘https://jsonplaceholder.typicode.com/users’)

Хаанаас татах гэж байгаа хаягаа тавьж өгнө.

.then(response => response.json())

Response буюу хариулт ирнэ. Хамгийн түрүүнд header ирнэ. Header амжилтай ирэх юм бол дараа body татдаг байж байгаа.

.then(data => this.setState({ robots: data }));

Data хүрээд ирэнгүүт нь төлөвруугаа хийнэ.

Component class удамшиж орж ирж байгаа болохоор урд талд заавалчгүй this гэж тавьж өгнө. This.setState гэж.