**Class Component**

import { Component } from "react";

class ClassComponent extends Component {

 state = {

      name: "João Victor",

      counter: 0

    };

  handlePClick = () => {

*this*.setState({ name: "Junior" });

  }

  handleAClick = () => {

    const { counter } = *this*.state;

*this*.setState({ counter: counter + 1 });

  };

  render() {

    const { name, counter } = *this*.state;

    return (

      <>

        <p *onClick*={*this*.handlePClick}>{name}</p>

        <a *onClick*={*this*.handleAClick}>{counter}</a>

      </>

    );

  }

}

export default ClassComponent;

**Array State**

import { Component } from "react";

class ArrayState extends Component {

  state = {

    posts: [

      {

        id: 1,

        title: "O titulo 1",

        body: "O corpo",

      },

      {

        id: 2,

        title: "O titulo 2",

        body: "O corpo 2",

      },

      {

        id: 3,

        title: "O titulo 3",

        body: "O corpo 3",

      },

    ],

  };

  render() {

    const { posts } = *this*.state;

    return (

      <>

        {posts.map((*post*) => (

          <div *key*={*post*.id}>

            <h1>{*post*.title}</h1>

            <p>{*post*.body}</p>

          </div>

        ))}

      </>

    );

  }

}

export default ArrayState;

**Lifecycles Methods**

import { Component } from "react";

class ArrayState extends Component {

  state = {

    counter: 0,

    posts: [

      {

        id: 1,

        title: "O titulo 1",

        body: "O corpo",

      },

      {

        id: 2,

        title: "O titulo 2",

        body: "O corpo 2",

      },

      {

        id: 3,

        title: "O titulo 3",

        body: "O corpo 3",

      },

    ],

  };

  timeoutUpdate = null;

  componentDidMount () {

*this*.handleTimeout()

  }

  componentDidUpdate () {

*this*.handleTimeout()

  }

  componentWillUnmount () {

      clearTimeout(*this*.timeoutUpdate)

  }

  handleTimeout = () => {

    const { posts, counter } = *this*.state;

    posts[0].title = 'O titulo Mudou'

*this*.timeoutUpdate = setTimeout(() =>{

*this*.setState({ posts, counter: counter + 1 })

    }, 2000)

  }

  render() {

    const { posts, counter } = *this*.state;

    return (

      <>

      <h1>{counter}</h1>

        {posts.map((*post*) => (

          <div *key*={*post*.id}>

            <h1>{*post*.title}</h1>

            <p>{*post*.body}</p>

          </div>

        ))}

      </>

    );

  }

}

export default ArrayState;

**Data Fetching JSX**

import { Component } from "react";

import "./DataFetching.css";

class ArrayState extends Component {

  state = {

    posts: [],

  };

  //with fetch and promises.

  //componentDidMount () {

  //    fetch('https://jsonplaceholder.typicode.com/posts')

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

  //     .then (posts => this.setState({ posts }))

  //  }

  componentDidMount() {

*this*.loadPosts();

  }

  loadPosts = async () => {

    const postsResponse = fetch("https://jsonplaceholder.typicode.com/posts");

    const photosResponse = fetch("https://jsonplaceholder.typicode.com/photos");

    const [posts, photos] = await *Promise*.all([postsResponse, photosResponse]);

    const postsJson = await posts.json();

    const photosJson = await photos.json();

    const postsAndPhotos = postsJson.map((*post*, *index*) => {

      return { ...*post*, cover: photosJson[*index*].url };

    });

*this*.setState({ posts: postsAndPhotos });

  };

  render() {

    const { posts } = *this*.state;

    return (

      <section *className*="container">

        <div *className*="posts">

          {posts.map((*post*) => (

            <div *className*="post">

              <img *src*={*post*.cover} *alt*={*post*.title} />

              <div *key*={*post*.id} *className*="post-content">

                <h1>{*post*.title}</h1>

                <p>{*post*.body}</p>

              </div>

            </div>

          ))}

        </div>

      </section>

    );

  }

}

export default ArrayState;

**Data fetching css**

\* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;

}

*.container* {

    min-height: 100vh;

    background: #eeeeee;

    padding: 30px;

}

*.posts* {

    display: grid;

    grid-template-columns: repeat(*auto-fill*, minmax(280px, 1fr));

    gap: 30px;

}

*.post* {

    background: #fff;

    box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);

    transition: transform 100ms ease-in-out;

}

*.post:hover* {

    transform: scale(1.05);

    background: linear-gradient(135deg, #07a7e3 0%, #32dac3 100%);

}

*.post* img {

    max-width: 100%;

}

*.post-content* {

    padding: 30px;

}