

STATES

- State is a plain js object used by react to represent an information about component's current situation.
- In functional based we use **useState**.
- useState: is a react hook that allows you to add states to the functional based components and it returns an array with two values, one is for initial value and one is for function to update initial value.

```
import React, { Fragment } from "react";
import { useState } from "react";
import JSON from "../data.json"

import ChildComponent from "../components/ChildComponent";

const ParentComponent = () => {
  const dataArray = [1, 2, 3, 4, 5];
  const [currentIndex, setCurrentIndex] = useState(0);

  const handleClick = () => {
    setCurrentIndex((currentIndex) => currentIndex + 1);
  };

  return (
    <div>
      <button onClick={handleClick}>Print Next</button>
      <ChildComponent data={dataArray} currentIndex={currentIndex} />
    </div>
  );
};

export default ParentComponent;
```

=====

```
const ParentComponent = () => {
  let [lang, setLang] = useState(["apple", "mango", "litchi"])
  return (
    <div>
      {
        lang.map((val, index) => {
          return <Fragment key={index}>
            <li>{val}</li>
          </Fragment>
        })
      }
    </div>
  );
};
```

```

        </Fragment>
      })
    }
  </div>
);
}

export default ParentComponent;
=====

```

```

const ParentComponent = () => {
  let[courses,setCourses]=useState([
    {
      course_id:1,
      course_name:"MERN",
      trainers:["Beth","Carol","David"],
      price:5000,
      branch:'KCH',
      course_details:
      {
        frontend:['html','css','js','react'],
        backend:['java','spring','mongoose']
      }
    },
    {
      course_id:2,
      course_name:"JAVA FULL STACK",
      trainers:["Carl","Daryl","Bella"],
      price:2000,
      branch:'HYD',
      course_details:
      {
        frontend:['html','css','js','react'],
        backend:['java','spring','mongoose']
      }
    }
  ]])

  return (
    <div>
      <section id="courseBlock">
        <article>
          <main>
            <header>
              <h2>LIST OF COURSES</h2>
            </header>
            <aside className="container">
              {
                courses.map((course)=>{

```

```

        return(
            <div className="list" key={course.id}>
                <h3>{course.course_name}</h3>
                <p><span>Branch: </span>{course.branch}</p>
                <p>Price: &#8377;{course.price}</p>
                <p>
                    <span>Trainers: </span>
                    <div
className="trainer">{course.trainers.map((trainer,index)=>{
                        return <Fragment key={index+1}>
                            <h5>{trainer}</h5>

                            </Fragment>
                        })}</div>
                </p>
                <span>FRONTEND TECHNOLOGIES: </span>
                <div className="c">

                    {

course.course_details.frontend.map((front,index)=>{
                        return(
                            <Fragment key={index+1}>
                                <p id="front">{front}</p>

                                </Fragment>
                            )

                        })
                    }
                </div>
                <span>BACKEND TECHNOLOGIES: </span>
                <div className="c">

                    {

course.course_details.backend.map((back,index)=>{
                        return(
                            <Fragment key={index+1}>
                                <p id="back">{back}</p>

                                </Fragment>
                            )

                        })
                    }
                </div>
            <div className="btn-group">

```

```

        <button>apply</button>
      </div>
    </div>

    )
  })
}
</aside>
</main>
</article>

</section>
</div>
);
}

export default ParentComponent;

```

BY USING JSON:

```

const ParentComponent = () => {
  let [courses, setCourses] = useState(JSON);

  return (
    <div>
      <section id="courseBlock">
        <article>
          <main>
            <header>
              <h2>LIST OF COURSES</h2>
            </header>
            <aside className="container">
              {
                courses.map((course) => {
                  return (
                    <div className="list" key={course.id}>
                      <h3>{course.course_name}</h3>
                      <p><span>Branch: </span>{course.branch}</p>
                      <p>Price: &#8377;{course.price}</p>
                      <p>
                        <span>Trainers: </span>
                        <div
                          className="trainer">{course.trainers.map((trainer, index) => {
                            return <Fragment key={index+1}>
                              <h5>{trainer}</h5>
                            </Fragment>
                          })}
                        </div>
                      </p>
                    </div>
                  )
                })
              }
            </aside>
          </main>
        </article>
      </section>
    </div>
  );
}

```

```

    }}</div>
  </p>
  <span>FRONTEND TECHNOLOGIES: </span>
  <div className="c">

    {

course.course_details.frontend.map((front,index)=>{
    return(
      <Fragment key={index+1}>
        <p id="front">{front}</p>

      </Fragment>
    )

    })
  }
</div>
  <span>BACKEND TECHNOLOGIES: </span>
  <div className="c">

    {

course.course_details.backend.map((back,index)=>{
    return(
      <Fragment key={index+1}>
        <p id="back">{back}</p>

      </Fragment>
    )

    })
  }
</div>
  <div className="btn-group">
    <button>apply</button>
  </div>
</div>

  )
})
}
</aside>
</main>
</article>

</section>
</div>

```

```

    );
}

export default ParentComponent;

```

```

=====

const ParentComponent = () => {
  let [count, setCount] = useState({
    count_No: 0,
    loading: false,
    name: "lori"
  })
  const Increment = () => {
    setCount({ count_No: count.count_No + 1, loading: true, name: "tenner" });
  }
  const Decrement = () => {
    setCount({ count_No: count.count_No - 1, loading: true, name: "patrick" });
  }
  const Reset = () => {
    setCount({ count_No: 0, loading: true, name: "edd" });
  }
  return (
    <div>
      <p>
        Increment/decrement by <strong>{count.name}</strong>
      </p>
      <h1>{count.count_No}</h1>
      <div className="btn-group">
        <button onClick={Increment}>Increment</button>
        <button onClick={Decrement}>Decrement</button>
        <button onClick={Reset}>Reset</button>
      </div>
    </div>
  )
}

export default ParentComponent

```

```

=====

const ParentComponent = () => {

  let [cart, setCart] = useState(0);
  let addToCart = () => {
    setCart(preValue => preValue + 100);
    setCart(preValue => preValue + 100);
  }

```

```
return (  
  <div>  
  
    <h2>CART: {cart}</h2>  
  
    <button onClick={addToCart}>addToCart</button>  
  
  </div>  
)  
}  
  
export default ParentComponent
```

=====