

# REACTJS - MAP- CRUD OPERATION

ANECO ACADEMY

## CONDITIONAL OPERATOR

**You can add conditional operator in JSX to make decision for output.**

**(condition)?true:false**

**Condition you can use the state variable to decide the true condition.**

**Inside true you can have a block of html code inserted in the first block**

**Inside false you can have block of html or null to show empty otherwise**

## CONDITIONAL OPERATOR EXAMPLE

```
return (  
  <div className="App">  
    <button onClick={getHandler}>get State</button>  
    <button onClick={updateHandler}>updated State</button>  
    <button onClick={switchHandler}>Submit</button>      You, seco  
    {getState.otherState?  
      <div>  
        <Product  
          click={nameChangeHandler}  
          name={getState.product[0].name} price={getState.product[0].price}/>  
        <Product  
          click={nameChangeHandler}  
          name={getState.product[1].name} price={getState.product[1].price}/>  
      </div>:null  
    }  
  </div>  
) ;
```

## LOOP INSIDE JSX

**To loop the data inside JSX we need to use map function  
Inside each render of map data we need to have key  
property**

**Key property can be index from map function or any unique  
key that can be generated from frontend or backend**

**If data is empty map will not executed anything**

**Map function support will array**

## Map function example

```
if(getState.otherState){  
  person=(<div>  
    {getState.product.map((product,index)=>{  
      return(<Product  
        key={index}  
        click={nameChangeHandler}  
        name={product.name}  
        price={product.price}/>  
      )  
    })}  
    </div>  
  )  
}
```

You, seconds ago • Uncommitted

## EVENT CALLED INSIDE MAP FUNCTION

**Any events can be called inside the map function as well.  
To differentiate between data we need to pass index value  
during event call.**

**Then you can call state to update for tht**