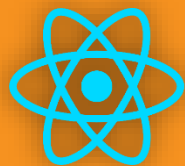




Școala
informală
de IT



React Components. Advanced

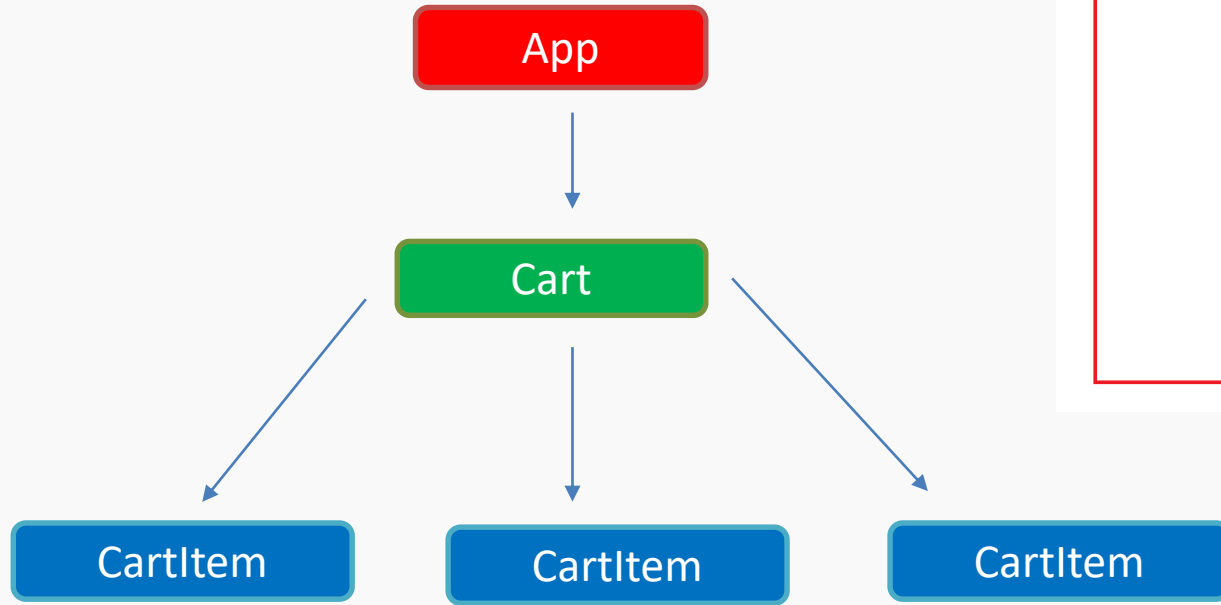


Agenda

- Pass data using props
- Events
- Multiple components
- Functional components
- Lifecycle methods. Hooks.



Components tree



App

Cart

CartItem

iphone 13	-	0	+
iphone 13	-	0	+
iphone 13	-	0	+
iphone 13	-	0	+

App Component

```
import React, { Component } from 'react';
import './App.css';
import Cart from './components/Cart';

class App extends React.Component {
  render() {
    return (
      <div className="App">
        <Cart />
      </div>
    );
  }
}

export default App;
```

Cart Component

```
import React, { Component } from 'react';
import CartItem from './CartItem';

export default class Cart extends Component {
  render() {
    return (
      <div>
        <CartItem />
        <CartItem />
        <CartItem />
        <CartItem />
      </div>
    );
  }
}
```

CartItem Component

```
import React, { Component } from 'react';

export default class CartItem extends
Component {
  state = { count: 0 };

  styles = {
    fontSize: 18,
  };

  handleIncrement = () => {
    this.setState({ count:
this.state.count + 1 });
  };

  handleDecrement = () => {
    if (this.state.count > 0)
this.setState({ count: this.state.count - 1
});
  };
};
```

```
getCounterClasses() {
  let classes = 'badge m-2 bg-';
  const { count } = this.state;
  classes += count === 0 ? 'warning' : 'info';
  return classes;
}

render() {
  return (
    <div>
      <span>iphone 13</span>
      <button onClick={this.handleDecrement}
className="btn btn-dark btn-sm">
        -
      </button>
      <span
className={this.getCounterClasses()}>{this.state.count}</spa
n>
      <button onClick={this.handleIncrement}
className="btn btn-dark btn-sm">
        +
      </button>
    </div>
  );
}
```

Props – Passing data to component

```
export default class CartItem extends Component {
  state = { count: 0 };

  render() {
    return (
      <div class="cart-item">
        <span>{this.props.productName}</span>
        <button onClick={this.handleDecrement} className="btn btn-dark btn-sm">
          -
        </button>
        <span className={this.getCounterClasses()}>{this.state.count}</span>
        <button onClick={this.handleIncrement} className="btn btn-dark btn-sm">
          +
        </button>
      </div>
    );
  }
}
```

```
export default class Cart extends Component {
  render() {
    return (
      <div>
        <CartItem productName={'iphone 12'} />
        <CartItem productName={'Lenor'} />
        <CartItem productName={'Samsung UHD'} />
        <CartItem productName={'Busuioaca de bohotin'} />
      </div>
    );
  }
}
```

iphone 12 - 0 +

Lenor - 0 +

Samsung UHD - 0 +

Busuioaca de bohotin - 0 +

Rendering list

```
export default class CartItem extends Component {
  state = { count: 0 };

  render() {
    return (
      <div class="cart-item">
        <span>{this.props.productName}</span>
        <button onClick={this.handleDecrement} className="btn btn-dark btn-sm">
          -
        </button>
        <span className={this.getCounterClasses()}>{this.state.count}</span>
        <button onClick={this.handleIncrement} className="btn btn-dark btn-sm">
          +
        </button>
      </div>
    );
  }
}
```

```
export default class Cart extends Component {
  products = [
    { id: 1, name: 'iphone 13' },
    { id: 2, name: 'Motorola V3' },
    { id: 3, name: 'Siemens A50' },
    { id: 4, name: 'Nokia 3310' },
  ];

  render() {
    return (
      <div>
        {this.products.map((product) => (
          <CartItem key={product.id} productName={product.name} />
        ))}
      </div>
    );
  }
}
```



Passing Children (1)

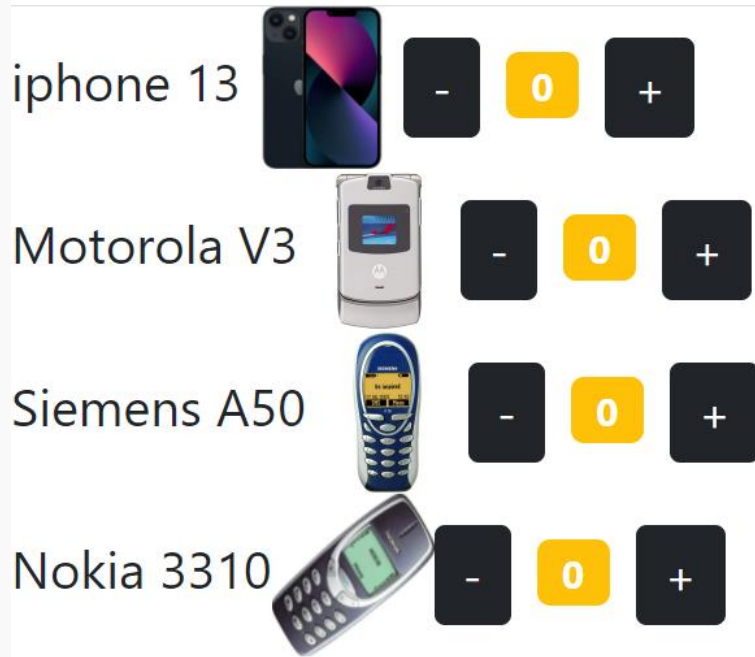
```
export default class Cart extends Component {
  products = [
    {
      id: 1,
      name: 'iphone 13',
      imageURL:
        'https://lcdn.altex.ro/resize/media/catalog/product/T/e/2bd48d28d1c32adea0e55139a4e6434a/Telefon_APPLE_iPhone_13_5G_128GB_Midnight_4_.jpg',
    },
    {
      id: 2,
      name: 'Motorola V3',
      imageURL:
        'https://s13emagst.akamaized.net/products/353/352715/images/img30067_05052006145051.jpg',
    },
    {
      id: 3,
      name: 'Siemens A50',
      imageURL:
        'https://www.manualdeinstruțiuni.ro/thumbs/products/1/238982-siemens-a50.jpg',
    },
    {
      id: 4,
      name: 'Nokia 3310',
      imageURL: 'https://m.media-amazon.com/images/I/81sQxjJBn1L.jpg',
    },
  ];
  render() {
    return (
      <div>
        {this.products.map((product) => (
          <CartItem key={product.id}>
            <CartProduct name={product.name} imageURL={product.imageURL} />
          </CartItem>
        ))}
      </div>
    );
  }
}
```



Passing Children (2)

```
export default class CartItem extends Component {  
  state = { count: 0 };  
  
  render() {  
    return (  
      <div class="cart-item">  
        {this.props.children}  
        <button onClick={this.handleDecrement} className="btn btn-dark btn-sm">  
          -  
        </button>  
        <span className={this.getCounterClasses()}>{this.state.count}</span>  
        <button onClick={this.handleIncrement} className="btn btn-dark btn-sm">  
          +  
        </button>  
      </div>  
    );  
  }  
}
```

```
export default class CartProduct extends Component {  
  render() {  
    return (  
      <>  
        <span>{this.props.name}</span>  
        <img style={{ width: 50, height: 50 }} src={this.props.imageUrl} />  
      </>  
    );  
  }  
}
```



Debugging react apps



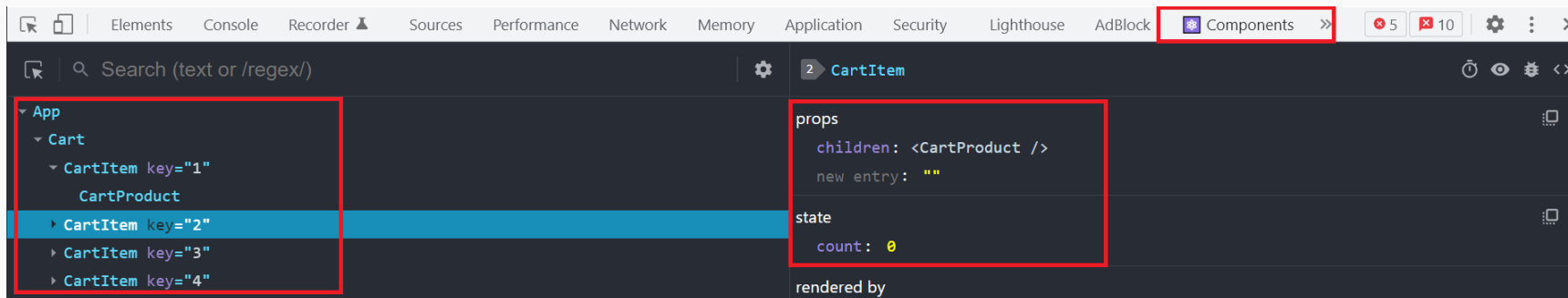
React Developer Tools

Offered by: Facebook

★★★★★ 1,372

Developer Tools

3,000,000+ users



The screenshot shows the React Developer Tools interface. The 'Components' tab is selected in the top bar. The left pane shows a tree view of the component structure, with 'App' expanded to show a 'Cart' component, which contains four 'CartItem' components with keys '1', '2', '3', and '4'. The 'CartItem key="2"' component is selected and highlighted in blue. The right pane shows the props and state for the selected 'CartItem' component. The props are `children: <CartProduct />` and `new entry: ""`. The state is `count: 0`. The 'rendered by' section is empty.

```
App
└─ Cart
  └─ CartItem key="1"
      CartProduct
  └─ CartItem key="2"
      CartProduct
  └─ CartItem key="3"
      CartProduct
  └─ CartItem key="4"
      CartProduct
```

```
CartItem
props
  children: <CartProduct />
  new entry: ""
state
  count: 0
rendered by
```

Props vs State



Props vs State



✓ props are read-only

✓ props can not be modified

✓ state changes can be asynchronous

✓ state can be modified using `this.setState`

Raising and handling events

Handle onDelete event

Cart

Raises onDelete event

CartItem

```
handleDelete = (productId) => {  
  // products are part of state object  
  const filteredProducts = this.state.products.filter(  
    (product) => product.id !== productId  
  );  
  this.setState({ products: filteredProducts });  
};  
  
render() {  
  return (  
    <div>  
      {this.state.products.map((product) => (  
        <CartItem  
          key={product.id}  
          onDelete={() => this.handleDelete(product.id)}  
        >  
          <CartProduct name={product.name} imageUrl={product.imageUrl} />  
        </CartItem>  
      ))}  
    </div>  
  );  
}
```

```
<button  
  onClick={this.props.onDelete}  
  className="btn btn-danger btn-sm m-2"  
>  
  Delete  
</button>
```