

16. Rendering List of Items in JSX

- Until now, we are just hardcoding the expense data

```
Expenses.js — react-complete-guide
src > components > Expenses > Expenses.js > Expenses
7  function Expenses(props) {
8    const [FilteredYear, setFilteredYear] = useState("2022");
9
10   const filterChangeHandler=(year)=>{
11     setFilteredYear(year);
12   }
13
14   return (
15     <div>
16       <Card className="expenses">
17         <ExpenseFilter selected={FilteredYear} onFilterChange={filterChange}
18         <ExpenseItem
19           title={props.items[0].title}
20           amount={props.items[0].amount}
21           date={props.items[0].date}
22         />
23         <ExpenseItem
24           title={props.items[1].title}
25           amount={props.items[1].amount}
26           date={props.items[1].date}
27         />
28         <ExpenseItem
29           title={props.items[2].title}
30           amount={props.items[2].amount}
31           date={props.items[2].date}
32         />
33       </Card>
34     </div>

```

- Instead of doing this, we want it to be dynamic
 - How can we render list of items dynamically in JSX?

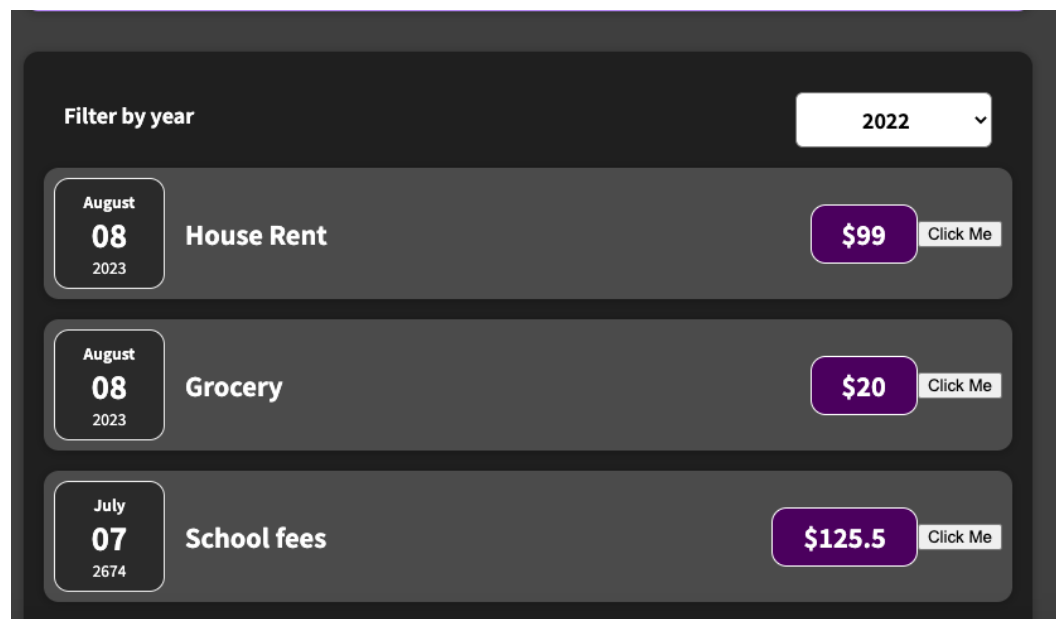
```
JavaScript Demo: Array.map()
1  const array1 = [1, 4, 9, 16];
2
3  // pass a function to map
4  const map1 = array1.map(x => x * 2);
5
6  console.log(map1);
7  // expected output: Array [2, 8, 18, 32]
8
```

- 1.
- 2.

```
Expenses.js — react-complete-guide
Expenses.css Expenses.js ExpenseDate.js ExpenseDate.css Card.js Card.css Expensesel
src > components > Expenses > Expenses.js > Expenses
13
14   return (
15     <div>
16       <Card className="expenses">
17         <ExpenseFilter selected={FilteredYear} onFilterChange={filterChangeHandler} ></ExpenseFilter>
18         {
19           <ExpenseItem title={props.items[0].title} amount={props.items[0].amount} date={props.items[0].date} />,
20           <ExpenseItem title={props.items[1].title} amount={props.items[1].amount} date={props.items[1].date} />,
21           <ExpenseItem title={props.items[2].title} amount={props.items[2].amount} date={props.items[2].date} />
22         }
23       </Card>
24     </div>
25   );
26 }
27
28
29 export default Expenses;
30
```

NOTE : in react, when a list of components are given in JSX, then it renders each of the component in the list side by side

- Here in line 18 we have a {}, this is for using js logic in JSX, here we are using js syntax for list, so we are using {}.
- And in line 19 we have a list, which has 3 components, so it renders it side by side.



- Now Using these 2 concepts, we can render the list of items dynamically

```

Expenses.js — react-complete-guide
src > components > Expenses > Expenses.js
16
17   return (
18     <div>
19       <Card className="expenses">
20         <ExpenseFilter selected={FilteredYear} onFilterChange={filterChangeHandler} ></ExpenseFilter>
21         {
22           // [<ExpenseItem title={props.items[0].title} amount={props.items[0].amount} date={props.items[0].date} />,
23           // <ExpenseItem title={props.items[1].title} amount={props.items[1].amount} date={props.items[1].date} />,
24           // <ExpenseItem title={props.items[2].title} amount={props.items[2].amount} date={props.items[2].date} /> ]
25         }
26
27         {props.items.map((expense) => <ExpenseItem title={expense.title} amount={expense.amount} date={expense.date} />)}
28
29       </Card>
30     </div>
31   );
32
33 }
34
35 export default Expenses;
36

```

○

```

src > components > Expenses > Expenses.js
37 // React.js.org/en/learn/react-devtools

props.items                               Expenses.js:14
▼ (3) [{...}, {...}, {...}] ⓘ
  ▶ 0: {id: 'e1', title: 'House Rent', amount: 99, date: Tue Aug 29 20
  ▶ 1: {id: 'e2', title: 'Grocery', amount: 20, date: Wed Aug 02 2023
  ▶ 2: {id: 'e3', title: 'School fees', amount: 125.5, date: Wed Jul 0
    length: 3
  ▶ [[Prototype]]: Array(0)

after map                                  Expenses.js:15
▼ (3) [{...}, {...}, {...}] ⓘ
  ▶ 0: {$$typeof: Symbol(react.element), key: null, ref: null, props:
  ▶ 1: {$$typeof: Symbol(react.element), key: null, ref: null, props:
  ▶ 2: {$$typeof: Symbol(react.element), key: null, ref: null, props:
    length: 3
  ▶ [[Prototype]]: Array(0)

```

○

- Here props.items is a list of expense item details
- After map function, it is creating an another list which contains a list of components
-