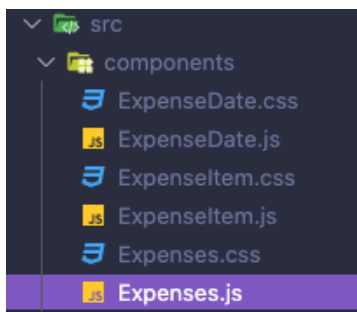


10. Splitting part-2

- Look at our app.js

```
3 function App() {
4   const expenses = [
5     {
6       id: 'e1',
7       title: 'Toilet Paper',
8       amount: 94.12,
9       date: new Date(2020, 7, 14),
10    },
11    { id: 'e2', title: 'New TV', amount: 799.49, date: new Date(2021, 2, 12) },
12    {
13      id: 'e3',
14      title: 'Car Insurance',
15      amount: 294.67,
16      date: new Date(2021, 2, 28),
17    },
18    {
19      id: 'e4',
20      title: 'New Desk (Wooden)',
21      amount: 450,
22      date: new Date(2021, 5, 12),
23    },
24  ];
25
26  return (
27    <div>
28      <h2>Let's get started!</h2>
29      <ExpenseItem
30        title={expenses[0].title}
31        amount={expenses[0].amount}
32        date={expenses[0].date}
33      />
34      <ExpenseItem
35        title={expenses[1].title}
36        amount={expenses[1].amount}
37        date={expenses[1].date}
38      />
```

- Our app.js looks a bit complex because we have reused the expenseitems several times, instead we can create a separate component which can have all the expenseitems and importing that here in app.js and using as a single custom components makes it simple
- In simple words we are going to nesting this expenseitem component into another component



```
src > components > Expenses.js > ...
1  import ExpenseItem from "../ExpenseItem";
2  import './Expenses.css'
3
4  function Expenses(props) {
5    return (
```

- If you notice, in our app.js, we have used the custom html element expenseitem along with some attributes which was extracted from an array
- But now we are going to use the expenseitem component inside another component now how could we have access to that array
- It can be done using props
- In app.js

```
src > App.js > App
23  return (
24    <div>
25      <h2>Let's get started!</h2>
26      <Expenses items={expenses}></Expenses>
27    </div>
28  );
29 }
30
31 export default App;
```

- We are passing the attributes
- It can be accessed using props in another component

```
src > components > Expenses.js > Expenses
1  import ExpenseItem from "../ExpenseItem";
2  import './Expenses.css'
3
4  function Expenses(props) {
5    return (
6      <div className="expenses">
7        <ExpenseItem
8          title={props.items[0].title}
9          amount={props.items[0].amount}
10         date={props.items[0].date}
11        ></ExpenseItem>
12
13        <ExpenseItem
14          title={props.items[1].title}
15          amount={props.items[1].amount}
16          date={props.items[1].date}
17        ></ExpenseItem>
```

- As items is an array, we are using the index value

- If you have noticed, you could have seen that we are enclosing the expenseitem into an div

```
src > components > Expenses.js > Expenses
5   return (
6     <div className="expenses">
7       <ExpenseItem
8         title={props.items[0].title}
9         amount={props.items[0].amount}
10        date={props.items[0].date}
11      ></ExpenseItem>
12
13      <ExpenseItem
14        title={props.items[1].title}
15        amount={props.items[1].amount}
16        date={props.items[1].date}
17      ></ExpenseItem>
18
19      <ExpenseItem
20        title={props.items[2].title}
21        amount={props.items[2].amount}
22        date={props.items[2].date}
23      ></ExpenseItem>
24
25      <ExpenseItem
26        title={props.items[3].title}
27        amount={props.items[3].amount}
28        date={props.items[3].date}
29      ></ExpenseItem>
30    </div>
```

-
- Because react component sends only one component but it may have nested elements inside it but at the outer it should be a single element
- In app.js

```
src > App.js > App
17
18   {title: 'Fuel Expense',
19     amount: 200,
20     date: new Date(2022, 3, 29)},
21
22   ]
23   return (
24     <div>
25       <h2>Let's get started!</h2>
26       <Expenses items={expenses}></Expenses>
27     </div>
28   );
29
30
31   export default App;
32
```

-
- Now our app.js looks simple