

9. Splitting larger components

- We have this component, which is quite larger, it is upto you to decide whether to split it or not
- But it is a good practice to split as much as you can

```
src > components > ExpenseItem.js > ExpenseItem > year
1  import './ExpenseItem.css';
2
3  function ExpenseItem(props) {
4    const day=props.date.toLocaleString("en-US", { day: "numeric" })
5    const month=props.date.toLocaleString("en-US", { month: "long" })
6    const year=props.date.toLocaleString("en-US", { year: "numeric" })
7    return (
8      <div className="expense-item">
9        <div>
10         <div>{month}</div>
11         <div>{year}</div>
12         <div>{day}</div>
13       </div>
14
15       <div className="expense-item__description">
16         <h2>{props.title}</h2>
17         <div className="expense-item__price">${props.amount}</div>
18       </div>
19     </div>
20   );
21 }
22
23 export default ExpenseItem;
24
```

○

- Here we could split the date related items into a separate component



•

- Now we could copy this and paste it into our new component

```
src > components > ExpenseItem.js > ExpenseItem
1  import './ExpenseItem.css';
2
3  function ExpenseItem(props) {
4      const day=props.date.toLocaleString("en-US"
5      const month=props.date.toLocaleString("en-US"
6      const year=props.date.toLocaleString("en-US"
7      return (
8          <div className="expense-item">
9              <div>
10                 <div>{month}</div>
11                 <div>{year}</div>
12                 <div>{day}</div>
13             </div>
14
15             <div className="expense-item__descripti
16                 <h2>{props.title}</h2>
17                 <div className="expense-item__price">
18             </div>
19         </div>
20     );
21 }
22
23 export default ExpenseItem;
```

-
- Then we could replace this with our new component by importing it
- Before that we must create that component

```
src > components > ExpenseDate.js > default
1  function ExpenseDate() {
2      const day = props.date.toLocaleString("en-US", { day: "numeric" });
3      const month = props.date.toLocaleString("en-US", { month: "long" });
4      const year = props.date.toLocaleString("en-US", { year: "numeric" });
5      return (
6          <div>
7              <div>{month}</div>
8              <div>{year}</div>
9              <div>{day}</div>
10         </div>
11     );
12 }
13
14 export default ExpenseDate;
```

-
- Here we must get the date object to extract the month,day and date
- When you look at the expenseltem component we have the props → [props={title:' ',date:' ',amount:' '} now we have to pass the date property to the new component that we have created
- We know that we could pass it as we pass the attributes in html element like this <ExpenseDate date=' '></ExpenseDate>

```

src > components > ExpenseItem.js > ExpenseItem
1  import './ExpenseItem.css';
2
3  import ExpenseDate from './ExpenseDate';
4
5  function ExpenseItem(props) {
6    const day=props.date.toLocaleString("en-US", { day:
7    const month=props.date.toLocaleString("en-US", { mon
8    const year=props.date.toLocaleString("en-US", { year
9    return (
10     <div className="expense-item">
11       <ExpenseDate date={props.date} ></ExpenseDate>
12       <div className="expense-item__description">
13         <h2>{props.title}</h2>
14         <div className="expense-item__price">${props.s
15       </div>
16     </div>
17   );
18 }
19
20 export default ExpenseItem;

```

- And also we should receive it in the component function as a parameter

```

src > components > ExpenseDate.js > ExpenseDate
1  function ExpenseDate(props) {
2    const day = props.date.toLocaleString("en-US", {
3    const month = props.date.toLocaleString("en-US", {
4    const year = props.date.toLocaleString("en-US", {
5    return (
6      <div>
7        <div>{month}</div>
8        <div>{year}</div>
9        <div>{day}</div>
10     </div>
11   );
12 }
13
14 export default ExpenseDate;
15

```

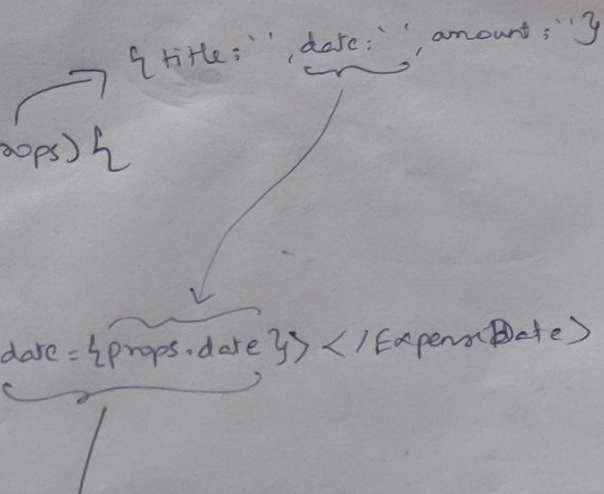
App.js

```
Expenses = [ { title: ' ', date: ' ', amount: ' ' },  
              {  
                {  
                  {  
                    {  
                      {  
                        {  
                          {  
                        {  
                      {  
                    {  
                  {  
                {  
              {  
            {  
          {  
        {  
      {  
    {  
  {  
}
```

```
function App() {  
  return (  
    <div>  
      <h2>  
        <ExpenseItem title={expenses[0].title}   
                      date={ } amount={ } > </ExpenseItem>  
        :  
        :  
      )  
    )  
  }  
}
```

ExpenseItem.js

```
function ExpenseItem(props) {  
  return (  
    <div>  
      <h2>  
        <ExpenseDate date={props.date} > </ExpenseDate>  
      )  
    )  
  }  
}
```



ExpenseDate.js

```
function ExpenseDate(props) {  
  return (  
    <div>  
      <h2>  
        <ExpenseDate date={props.date} > </ExpenseDate>  
      )  
    )  
  }  
}
```

- This is how the data is forwarded to components