

28. Rendering list contents based on a filter or condition

- First we need to create a input field to enter the year
- Let us create a separate component for that
- Create ExpenseFilter.js

```
src > components > Expenses > ExpenseFilter.js > ExpenseFilter
1  import React from "react";
2  import "../ExpenseFilter.css";
3
4  const ExpenseFilter = () => {
5    return (
6      <div className="expenses-filter">
7        <div className="expenses-filter__control">
8          <label>Filter by year</label>
9          <select>
10             <option value="2023">2023</option>
11             <option value="2022">2022</option>
12             <option value="2021">2021</option>
13             <option value="2020">2020</option>
14           </select>
15         </div>
16       </div>
17     );
18   };
19
20   export default ExpenseFilter;
```

○

- Now export this to Expenses.js

```
src > components > Expenses > Expenses.js > Expenses
2
3  import ExpenseItem from "../ExpenseItem";
4  import "../Expenses.css";
5  import "../ExpenseFilter";
6  import Card from "../UI/Card";
7  import ExpenseFilter from "../ExpenseFilter";
8
9  const Expenses=(props)=> {
10    return (
11      <Card className="expenses">
12        <ExpenseFilter></ExpenseFilter>
13        {props.items.map((singleexpense)=>
14          <ExpenseItem key={singleexpense.id} title={singleexpense.title}>
15          </ExpenseItem>
16        )}
17      </Card>
18    );
19  }
20
21
22   export default Expenses;
```

○

- Here in Expenses.js we don't need to render the entire list instead we could only display based on the year filter
- So here we must get the user's year input, after getting that we can filter it, but how could we get that values here in Expenses.js from ExpenseFilter.js
 - In ExpenseFilter.js let us get the value there itself
 - Let us add a listener here

```
src > components > Expenses > ExpenseFilter.js >
1  import React from "react";
2  import "../ExpenseFilter.css";
3
4  const ExpenseFilter = () => {
5    return (
6      <div className="expenses-filter">
7        <div className="expenses-filter__control">
8          <label>Filter by year</label>
9          <select onChange={()=>{}}>
10             <option value={"2023"}>2023</option>
11             <option value={"2022"}>2022</option>
12             <option value={"2021"}>2021</option>
13             <option value={"2020"}>2020</option>
14          </select>

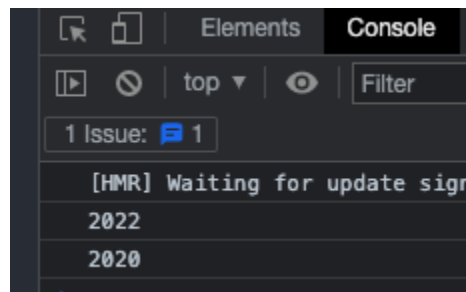
```

-
- Now let us create a handler function

```
src > components > Expenses > ExpenseFilter.js > ExpenseFilter >
1  import React from "react";
2  import "../ExpenseFilter.css";
3
4  const ExpenseFilter = () => {
5    const onChangeHandler=(event)=>{
6      console.log(event.target.value);
7    }
8    return (
9      <div className="expenses-filter">
10        <div className="expenses-filter__control">
11          <label>Filter by year</label>
12          <select onChange={onChangeHandler}>
13            <option value={"2023"}>2023</option>
14            <option value={"2022"}>2022</option>
15            <option value={"2021"}>2021</option>
16            <option value={"2020"}>2020</option>
17          </select>

```

-
- The onChange function passes in an event argument
- Here we accessed the value

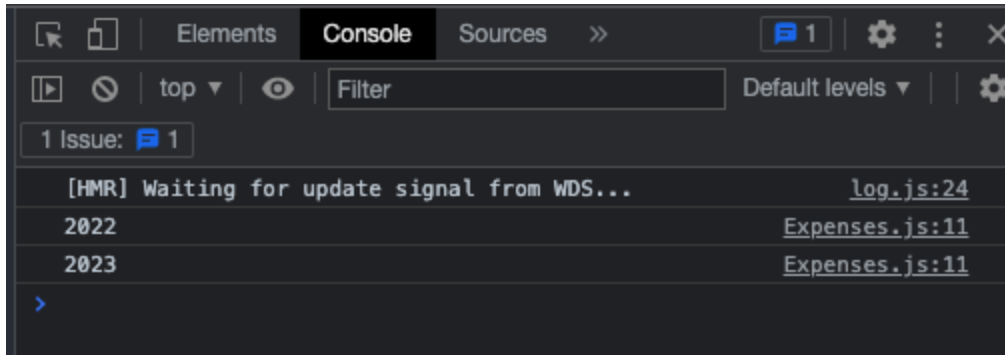


- Now in Expenses.js we want to access this selected year, so recall how we passed data from child to parent
- So in Expenses.js let us do this

```
src > components > Expenses > Expenses.js > Expenses > changeFilterhandler
2
3   import ExpenseItem from "../ExpenseItem";
4   import './Expenses.css';
5   import './ExpenseFilter';
6   import Card from "../UI/Card";
7   import ExpenseFilter from './ExpenseFilter';
8
9   const Expenses=(props)=> {
10     const changeFilterhandler=(selectedYear)=>{
11       console.log(selectedYear);
12     }
13     return (
14
15       <Card className=" (JSX attribute) onChangeFilter: (selectedYear:
16         <ExpenseFilter onChangeFilter={changeFilterhandler} ></ExpenseF
17         {props.items.map((singleexpense)=>
18           <ExpenseItem key={singleexpense.id} title={singleexpense.title}
19         )}
20       </Card>
21     );
22   }
23
24
25   export default Expenses;
```

-
- Now we can call this function inside ExpenseFilter.js along with the selctedyear by passing it as an parameter and we receive that parameter data in Expenses.js
- Now in ExpenseFilter.js

```
src > components > Expenses > ExpenseFilter.js > ExpenseFilter > onCl
1   import React from "react";
2   import './ExpenseFilter.css';
3
4   const ExpenseFilter = (props) => {
5     const onChangeHandler=(event)=>{
6       //console.log(event.target.value);
7       props.onChangeFilter(event.target.value)
8     }
9     return (
10       <div className="expenses-filter">
11         <div className="expenses-filter__control">
12           <label>Filter by year</label>
13           <select onChange={onChangeHandler}>
14             <option value={"2023"}>2023</option>
15             <option value={"2022"}>2022</option>
16             <option value={"2021"}>2021</option>
17             <option value={"2020"}>2020</option>
18           </select>
19         </div>
20       </div>
21     );
```



- Now it is logged from Expenses.js
- Okay fine, now we got the entered year
- Now let us create a statevariable to hold the year the user entered in ExpenseFilter.js

```
src > components > Expenses > Expenses.js > Expenses
1  import React,{useState} from 'react';
2
3  import ExpenseItem from "../ExpenseItem";
4  import './Expenses.css';
5  import './ExpenseFilter';
6  import Card from "../UI/Card";
7  import ExpenseFilter from './ExpenseFilter';
8
9  const Expenses=(props)=> {
10     const [filteredYear,setFilteredYear]=useState('2021');
11
12     const changeFilter=(parameter) selectedYear: any
13     setFilteredYear(selectedYear);
14 }
15
16 return (
17
18     <Card className="expenses">
19         <ExpenseFilter onChangeFilter={changeFilterhandler} ><
20             {props.items.map((singleexpense)=>
21                 <ExpenseItem key={singleexpense.id} title={singleexpen
22             }
23
```

- Now in the changefilterhandler we used the setFilteredYear so that whenever user selects different year then this component will be re-evaluated

- Now its time to render only the items which is in that particular year
 - Now let us filter the list

```
src > components > Expenses > Expenses.js > Expenses
7  import ExpenseFilter from './ExpenseFilter';
8
9  const Expenses=(props)=> {
10   const [filteredYear,setFilteredYear]=useState('2021');
11
12   const changeFilterhandler=(selectedYear)=>{
13     setFilteredYear(selectedYear);
14   }
15   const filteredExpenses=props.items.filter((expense)=>{
16     return expense.date.getFullYear().toString() === filteredYear;
17   });
18
19   return (
20
21     <Card className="expenses">
22       <ExpenseFilter onChangeFilter={changeFilterhandler} ></ExpenseFilter>
23       {props.items.map((singleexpense)=>
24         <ExpenseItem key={singleexpense.id} title={singleexpense.title
25
26
```

-
- Now in line 23, instead of retrieving items from the entire expense list we could use the filterExpenses list

```
src > components > Expenses > Expenses.js > Expenses
7  import ExpenseFilter from './ExpenseFilter';
8
9  const Expenses=(props)=> {
10   const [filteredYear,setFilteredYear]=useState('2021');
11
12   const changeFilterhandler=(selectedYear)=>{
13     setFilteredYear(selectedYear);
14   }
15   const filteredExpenses=props.items.filter((expense)=>{
16     return expense.date.getFullYear().toString() === filteredYear;
17   });
18
19   return (
20
21     <Card className="expenses">
22       <ExpenseFilter onChangeFilter={changeFilterhandler} ></ExpenseFilter>
23       {filteredExpenses.map((singleexpense)=>
24         <ExpenseItem key={singleexpense.id} title={singleexpense.title} amount={singl
25
26
27       </Card>
28     );
29   }
30
31   export default Expenses;
```

- Now we are ready

Title

Amount

Date

dd/mm/yyyy

Add Expense

Filter by year

2021

April
2021
3

College Fees

\$500

- We have used state variable for filteredYear in Expenses.js, if we pass that state variable to ExpenseFilter component then that would also re-evaluate, so lets do it

```

src > components > Expenses > Expenses.js > Expenses
17   });
18
19   return (
20
21     <Card className="expenses">
22       <ExpenseFilter onChangeFilter={changeFilterhandler} selected={filteredYear} ></ExpenseFilter>
23       {filteredExpenses.map((singleexpense)=>
24         <ExpenseItem key={singleexpense.id} title={singleexpense.title} amount={singleexpense.amount} da
25       )}
26     </Card>
27   );
28
29 }

```

```
src > components > Expenses > ExpenseFilter.js > ExpenseFilter
1  import React from "react";
2  import "./ExpenseFilter.css";
3
4  const ExpenseFilter = (props) => {
5    const onChangeHandler=(event)=>{
6      //console.log(event.target.value);
7      props.onChangeFilter(event.target.value)
8    }
9    return (
10     <div className="expenses-filter">
11       <div className="expenses-filter__control">
12         <label>Filter by year</label>
13         <select onChange={onChangeHandler} value={props.selected}>
14           <option value={"2023"}>2023</option>
15           <option value={"2022"}>2022</option>
16           <option value={"2021"}>2021</option>
17           <option value={"2020"}>2020</option>
18         </select>
19       </div>
20     </div>
21   );
22 };
23
24 export default ExpenseFilter;
25
```