15. Passing props from Child to parent component.

- Now, whenever a user add new expense using the form, we need to add it to the expenses json objects, but here the form is in ExpenseForm component and the json is in App Component
- Here App is parent, and it has one intermediary component and then the ExpenseForm Component
  - ExpenseForm.js

```
return (
  <form onSubmit={submitHandler}>
    <div className="new-expense__controls">
      <div className="new-expense__control">
        <label>Title</label>
        <input type="text" value={EnteredTitle} onChange={titleChangeHandler} />
      </div>
      <div className="new-expense__control">
        <label>Amount</label>
        <input
          type="number"
          min="0.01"
          step="0.01"
          onChange={amountChangeHandler}
          value={EnteredAmount}
        />
      </div>
      <div className="new-expense__control">
        <label>Date</label>
        <input
```

  - App.js

```
import Expenses from './components/Expenses/Expenses';
import NewExpense from './components/NewExpense/NewExpense'

function App() {
  const expenses = [
    { id:"e1", title: "House Rent", amount: 99.0, date: new Date(2023, 7, 29) },
    { id:"e2", title: "Grocery", amount: 20, date: new Date(2023, 7, 2) },
    { id:"e3", title: "School fees", amount: 125.5, date: new Date(2023, 7818) },
  ];
  return (
    <div>

      <NewExpense></NewExpense>
      <Expenses items={expenses} />
    </div>
  );
}

export default App;
```

- Now when a user submits the form, the entered data must go to the App component

- To pass some value from parent to child, we can use props as we saw earlier
- To pass some value from child to parent component, we can use props but in a little different way
    - App → NewExpense → ExpenseForm
    - First let us receive the entered data in NewExpense component from the child component(ExpenseForm)
        - So in parent component(NewExpense) do the following

-
    - Here while calling the child component ExpenseForm, let us pass a prop which is onSaveExpenseData which is assigned to function
    - That function onSaveEXPENSEDATAHANDLER accepts an argument
    - In line 10, we are just passing the name of the function not executing it
- Now in child component, do the following

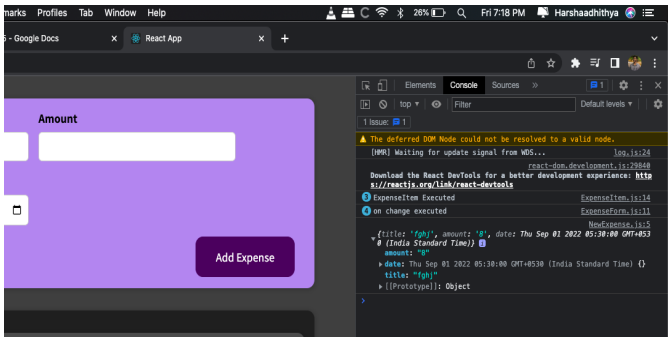
    -
        - Accept the props argument, because now we are passing the props

- ExpenseForm.js — react-complete-guide

App.js | ExpenseItem.js | NewExpense.js | ExpenseForm.js | Expense

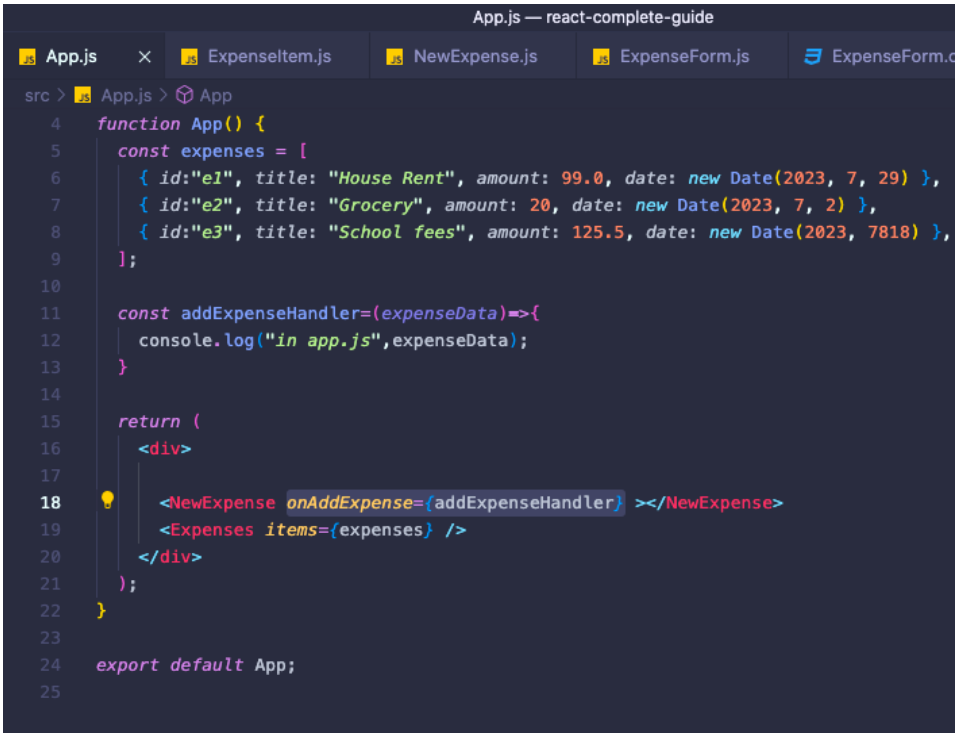
src > components > NewExpense > ExpenseForm.js > ExpenseForm > submitHandler

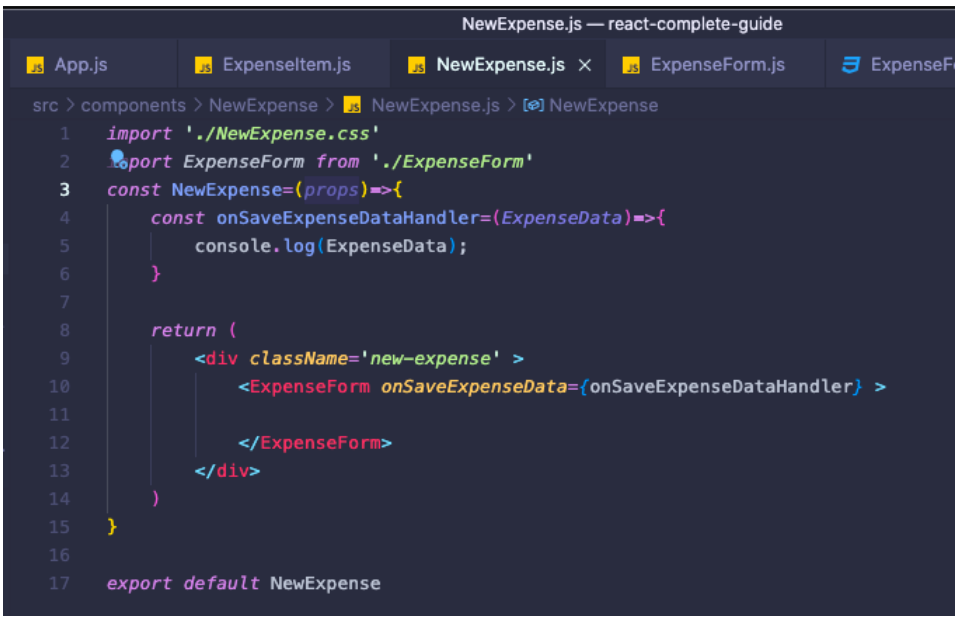
```
};

const submitHandler=(event)=>{
  event.preventDefault();
  const ExpensData={
    title:EnteredTitle,
    amount:EnteredAmount,
    date:new Date(EnteredDate)
  }
  setEnteredTitle('');
  setEnteredAmount('');
  setEnteredDate('');

  // console.log(ExpensData);
  props.onSaveExpenseData(ExpensData);
}

return (
  <form onSubmit={submitHandler}>
    <div className="new-expense__controls">
      <div className="new-expense__control">
        <label>Title</label>
```

    - Now in onsubmithandler, instead of printing, let us call the function in parent component using props.onSaveExpenseData() and passing in the entered values.
  - Output:
    - 

    - Now we are getting the entered valuesfrom ExpenseForm component(child) to NewExpense component(parent)

- **Summary:**
  - **Here in parent component we are defining the function which accepts some argument**
  - **And then we are passing the name of that function as props to the child component**
  - **And in the child component we are accessing that function of parent component using props, and calling it and passing the values as argument to it, in child component that function is executed eventhough the definition of the function is in parent component, this is because we are just passing the pointer of that function from parent to child using props, so by accessing the pointer in child component we can make that function executable in it and also we can pass the data**

- Now we can do the similar to pass from NewExpense to App component
  - In parent App.js
    - 

    - Here the addExpensehandler accepts an argument and we are passing the pointer of the function as props to child component
  - In child component
    - 

      - Accepting props

NewExpense.js — react-complete-guide

App.js | ExpenseItem.js | NewExpense.js | ExpenseForm.js | ExpenseForm

src > components > NewExpense > NewExpense.js > NewExpense > onSaveExpenseDataHandler

```
1  import './NewExpense.css'
2  import ExpenseForm from './ExpenseForm'
3  const NewExpense=(props)=>{
4      const onSaveExpenseDataHandler=(ExpenseData)=>{
5          // console.log(ExpenseData);
6          props.onAddExpense(ExpenseData);
7      }
8
9      return (
10         <div className='new-expense' >
11             <ExpenseForm onSaveExpenseData={onSaveExpenseDataHandler} >
12
13             </ExpenseForm>
14         </div>
15     )
16 }
17
18 export default NewExpense
```

- Now istead of printing, let us call the function defined in parent component from the child component using props and passing in the data

- Output: