```
// ============ New Component create and (props in multilavel)  ==========
/*

  NEW tow file created inside src folder
  1) ExpenseDate.js
  2) ExpenseDate.CSS

  Following Naming convension

  // component call one by one pass the (data or props)

  TREE

  index.js => App.js => ExpencesItem.js => ExpencesDate.js

  return back the html setted formant from the each component

*/

// =============== index.js START =============

import React from 'react';
import ReactDOM from 'react-dom';
import App from './App';

ReactDOM.render(<App />, document.getElementById('root'));

//******************** END *********************

//================== App.js START =================

import './App.css';
// import the Custom Component file
import ExpencesItem from './component/ExpencesItem';
function App() {

  const expenseArr = [
    // thsi array contain some object
    {
      id: 'e1',
      title: "Car Buy",
      price: 300000,
      date: new Date(2020, 3, 5),
    },
    {
      id: 'e2',
      title: "Book Buy",
      price: 300,
      date: new Date(2021, 4, 6),
    },
    {
      id: 'e3',
      title: "Phone Buy",
```

```jsx
54        price: 3000,
55        date: new Date(2023, 1, 3),
56      },
57
58    ];
59
60    return (
61      <div >
62        <h2 className="bg">When custom function render that time pass value by Atribute</h2>
63  {/* 1) inside the call component pass atributes then the props set the html code value by javascript
64      2) the pass key_name should be recieved excatly same name
65      3) title = {expenseArr[0].title} => key_name = value_obtain_form_object here
66  */}
67        <ExpencesItem
68          title={expenseArr[0].title}
69          price={expenseArr[0].price}
70          date={expenseArr[0].date}
71        ></ExpencesItem>
72        <ExpencesItem
73          title={expenseArr[1].title}
74          price={expenseArr[1].price}
75          date={expenseArr[1].date}
76        ></ExpencesItem>
77        <ExpencesItem
78          title={expenseArr[2].title}
79          price={expenseArr[2].price}
80          date={expenseArr[2].date}
81        ></ExpencesItem>
82      </div>
83    );
84  }
85
86  /* ****** output *********
87
88  2020-04-04T18:00:00.000Z
89  Car Buy
90  $300000
91
92  2021-05-05T18:00:00.000Z
93  Book Buy
94  $300
95
96  2023-02-02T18:00:00.000Z
97  Phone Buy
98  $3000
99  */
100 export default App;
101
102 //**************** END ********************
103
104 //=============== ExpencesItem.js START ============
105
106 // import CSS file
107 import './ExpencesItem.css';
```

```
108   // import the Date component file
109   import ExpenseDate  from './ExpenseDate';
110   function ExpencesItem(props) {
111     // using some js
112
113     return (
114       <div className="container">
115         <div className="date-item">
116           <ExpenseDate date = {props.date}> </ExpenseDate>
117         </div>
118         <div className="name-item">{props.title}</div>
119         <div className="price-item">${props.price}</div>
120
121       </div>
122     );
123   }
124
125   export default ExpencesItem;
126
127   //***************** END *************************
128
129   //=============== ExpencesDate.js START ============
130
131
132   // CSS file import for use this....
133   import './ExpenseDate.css';
134   // we can use CSS on this file
135
136   function ExpenseDate(props) {
137     // use some javascript code..
138     // date varibale should be base on the given (props vaibale) name
139
140     const month = props.date.toLocaleString('en-US', { month: 'long' });
141     const day = props.date.toLocaleString('en-US', { day: '2-digit' });
142     const Year = props.date.getFullYear();
143
144     return (
145
146       <div>
147         <div> {month} </div>
148         <div>{Year}</div>
149         <div>{day}</div>
150       </div>
151     );
152
153   }
154   export default ExpenseDate;
155
156   //**************** END ***********************
157
158   //***************** EXpencesItem.css **********
159     /*
160     *{
161       margin: 0;
```

```css
162        padding: 0;
163    }
164    .container{
165        width: 50%;
166        height: 300px;
167        margin: 0 auto;
168    }
169    .heading{
170        font-size: 25px;
171        background-color: gold;
172    }
173
174    /* we can use media query for verity devices */
175
176 //******************** END ********************
```