

```

1 // ===== Add Event listener to component (onClick) =====
2 /*
3     ***** Dynamically change view that time State *****
4     1) we need React STATE coz React only one time render but we the user want to
5     2) change some value of that time STATE help to change Dynamically
6     3) React does not care about the variables change or update After Reandering One time
7     4) IF we want check where the value change FORCE to React by using library method
8     5) import React , { useState } from 'react'
9     6) useState(default) will help to change to see.. it is useState one kind => hoock
10    7) const [title, setTitle] = useState(pops.title)
11    8) [title_variable, setTitle_function] when setTitle function call
12    the React Render Again the component
13    9) Compar the Old one and Set the New Value ans show
14
15    Add button on Expenseltem.js file and A Evnent Listener
16    1) not write so much code inside the JSX syntax
17    2) create the instraction function excatly below the componenet function
18    3) How to control the update BUTTON React (-_-)
19    4) inside button onClick = (take function value only)
20    5) <button onClick = {()}=> { console.log("Chicked")}> Click Me </button>
21    6) But the function we have write below the componenet is the conversion
22    7)
23
24 */
25
26 // ===== index.js START =====
27
28 import React from 'react';
29 import ReactDOM from 'react-dom';
30 import App from './App';
31
32 ReactDOM.render(<App />, document.getElementById('root'));
33
34 //***** END *****
35
36 //===== App.js START =====
37
38 import './App.css';
39 // import the Custom Component file
40 import ExpencesItem from './component/ExpencesItem';
41 function App() {
42
43     const expenseArr = [
44         // thsi array contain some object
45         {
46             id: 'e1',
47             title: "Car Buy",
48             price: 300000,
49             date: new Date(2020, 3, 5),
50         },
51         {
52             id: 'e2',
53             title: "Book Buy",
54             price: 200

```

```

54     price: 300,
55     date: new Date(2021, 4, 6),
56   },
57   {
58     id: 'e3',
59     title: "Phone Buy",
60     price: 3000,
61     date: new Date(2023, 1, 3),
62   },
63 ];
64
65
66 return (
67   <div >
68     <h2 className="bg">When custom function render that time pass value by Atribute</h2>
69   /* 1) inside the call component pass atributes then the props set the html code value by javascript
70      2) the pass key_name should be recieved excatly same name
71      3) title = {expenseArr[0].title} => key_name = value_obtain_form_object here
72   */
73   <ExpencesItem
74     title={expenseArr[0].title}
75     price={expenseArr[0].price}
76     date={expenseArr[0].date}
77   ></ExpencesItem>
78   <ExpencesItem
79     title={expenseArr[1].title}
80     price={expenseArr[1].price}
81     date={expenseArr[1].date}
82   ></ExpencesItem>
83   <ExpencesItem
84     title={expenseArr[2].title}
85     price={expenseArr[2].price}
86     date={expenseArr[2].date}
87   ></ExpencesItem>
88   </div>
89 );
90 }
91
92 /* ***** output *****
93
94 2020-04-04T18:00:00.000Z
95 Car Buy
96 $300000
97
98 2021-05-05T18:00:00.000Z
99 Book Buy
100 $300
101
102 2023-02-02T18:00:00.000Z
103 Phone Buy
104 $3000
105 */
106 export default App;
107
108 //***** END *****

```

```

109
110 //===== ExpencesItem.js START =====
111
112
113 // here useState hooks use to update change of variable
114
115 import React, { useState } from 'react';
116
117 // import CSS file
118 import './ExpencesItem.css';
119 // import the Date component file
120 import ExpenseDate from './ExpenseDate';
121
122 function ExpencesItem(props) {
123   // Here the onClick function Body
124
125   // let title = props.title;
126
127   // Here Array [variable, function] = useState(take_default_variable)
128
129   const [title, setTitle] = useState(props.title);
130
131   const clickHandler = () => {
132
133     //title = "Update Title";
134
135     // Here value change by function
136     setTitle("Update");
137
138     /* only Set the title will not show to website
139     When the Component Call That time Read At a time but => Title variable
140     After button Click How we can Set and Update value it is Our concern
141     */
142     console.log(title);
143   }
144
145   return (
146     <div className="container">
147       <div className="date-item">
148         <ExpenseDate date={props.date}> </ExpenseDate>
149       </div>
150       <div className="name-item">{title /* set variable */}</div>
151       <div className="price-item">${props.price}</div>
152       /* <button onClick = {() => { console.log("Clicked")}}> Clicked Me</button> */
153       <button onClick={clickHandler}> Change title </button>
154     /*
155       1) Calling not use the ()
156       2) COZ we just point the function Only
157       3) if We use (), When the this Item component call the the line function execute that time instantly
158       4) But we want to execute when the user click the button got it??
159     */
160   </div>
161 );
162 }

```

```

163
164 export default ExpencesItem;
165
166 //***** END *****
167
168 //===== ExpencesDate.js START =====
169
170
171 // CSS file import for use this....
172 import './ExpenseDate.css';
173 // we can use CSS on this file
174
175 function ExpenseDate(props) {
176     // use some javascript code..
177     // date varibale should be base on the given (props vaibale) name
178
179     const month = props.date.toLocaleString('en-US', { month: 'long' });
180     const day = props.date.toLocaleString('en-US', { day: '2-digit' });
181     const Year = props.date.getFullYear();
182
183     return (
184
185         <div>
186             <div> {month} </div>
187             <div>{Year}</div>
188             <div>{day}</div>
189         </div>
190     );
191
192 }
193 export default ExpenseDate;
194
195 //***** END *****
196
197 //***** EXpencesItem.css *****
198 /*
199 *{
200     margin: 0;
201     padding: 0;
202 }
203 .container{
204     width: 50%;
205     height: 300px;
206     margin: 0 auto;
207 }
208 .heading{
209     font-size: 25px;
210     background-color: gold;
211 }
212
213 /* we can use media query for verity devices */
214
215 //***** END *****

```