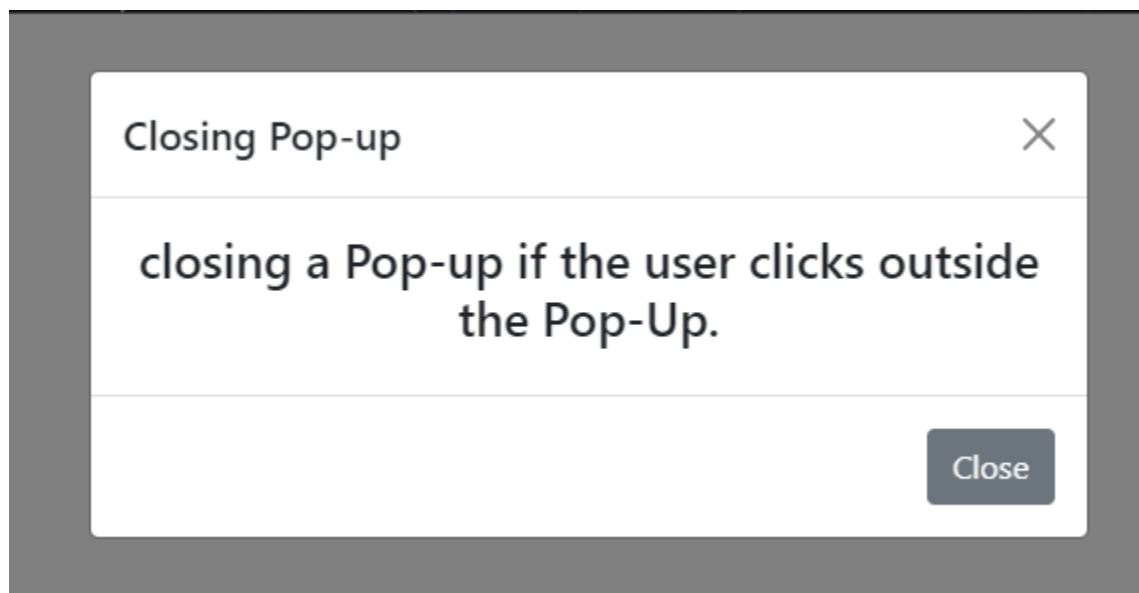


Q1:-Write function for hook to consume custom events related to Document or Window closing a Pop-up if the user clicks outside the Pop-Up.

Some Demo of How Look It's & how work all Function:-

Closing outside the Pop-Up demo

Output:- (anywhere Click Close its)



Q2 & Q3:-

Write an example of Interceptor to modify HTTP requests. Ensure that each field/parameter is trimmed before request is processed.

Write an example to Interceptor to modify response received from any API's before it is sent to client.

Code :-

```
src > JS index.js > ...
 7   import axios from "axios";
 8   |
 9   // For GET requests
10   axios.interceptors.request.use(
11     (req) => {
12       // Add configurations here
13       return req;
14     },
15     (err) => {
16       return Promise.reject(err);
17     }
18   );
19
20   // For POST requests
21   axios.interceptors.response.use(
22     (res) => {
23       // Add configurations here
24       if (res.status === 201) {
25         console.log("Posted Successfully");
26       }
27       return res;
28     },
29     (err) => {
30       return Promise.reject(err);
31     }
32   );
```

```
JS App.js    JS AxiosInterceptors.js 1 X    JS index.js

src > Assig-No-2&3 > JS AxiosInterceptors.js > ...
1  import React, { useEffect, useState } from "react";
2  import axios from "axios";
3
4  function AxiosInterceptors() {
5    const [data, setData] = useState(null);
6    const [val, setVal] = useState("");
7    const [fetchData, setFetch] = useState(false);
8
9    useEffect(() => {
10     if (fetchData) {
11       const payload = {
12         method: "POST",
13         body: JSON.stringify({ title: val }),
14       };
15       axios
16         .post("https://jsonplaceholder.typicode.com/posts", payload)
17         .then((res) => setData(res.data.id));
18     }
19   }, [fetchData]);
20
21   return (
22     <div>
23       {data && <h1>Your data is saved with Id: {data}</h1>}
24       <input
25         placeholder="Title of Post"
26         value={val}
27         onChange={(e) => setVal(e.target.value)}
28       />
29       <button onClick={() => setFetch(true)}>Save Data</button>
30     </div>
31   );
32 }
33
34 export default AxiosInterceptors;
35
```

Q4:-Write an example of custom Higher Order Component to make its children center aligned.

Output:-

Preferred Locations

Name: Ojo

Zone: Lagos State

Region: South West

Name: Ahiazu Mbaise

Zone: Imo State

Region: South East

Name: Akoko-Edo

Zone: Edo State

Region: South South

Name: Anka

Zone: Zamfara State

Functionality work:-




Preferred Locations

Name: Ahiazu Mbaise

Zone: Imo State

Region: South East



Q5:- Write a custom function that lets you deep compare dependencies and then re-render if any changes are detected.

Sol Code:-

```
JS App.js X JS DeepCompare.js X JS index.js
src > Assig-No-5 > JS DeepCompare.js > [🔍] DeepCompare
1  import { memo, useCallback, useEffect, useMemo, useState } from "react";
2
3  const List = memo((list) => {
4    useEffect(() => {
5      console.log("list changed");
6    }, [list]);
7
8    return <ul>{list?.length > 0 && list.map((e) => <li>{e}</li>)}</ul>;
9  });
10
11  const DeepCompare = () => {
12    const [a, setA] = useState(0);
13
14    const someFunc = useCallback(
15      () => console.log("This is a random function"),
16      []
17    );
18
19    useEffect(() => {
20      console.log("Use effect of someFunc's called");
21    }, [someFunc]);
22
23    const fakeList = useMemo(() => ["number 1", "number 2"], []);
24
25    return (
26      <div className="App">
27        <h1>Variable a: {a} </h1>
28        <button onClick={() => setA((v) => v + 1)}>Increase a</button>
29        <button onClick={someFunc}>call someFunc()</button>
30        <List list={fakeList} />
31      </div>
32    );
33  };
34
35  export default DeepCompare;
36
```

Q6:- Write an example React context and provider.

Output:-

Context API

This is A Child component

Christopher Nplan

This is Child B component

Intestellar

Code:-

```
10  function App() {
11    return (
12      <div className="App">
13        |
14        |
15        <h1>Context API</h1>
16        <MoviesContext>
17          <ChildAcom />
18          <ChildBcom />
19        </MoviesContext>
20      </div>
21    );
22  }
23
24  export default App;
25
```

```
JS App.js × JS MoviesContext.js × JS index.js
src > Assig-No-6 > JS MoviesContext.js > ...
1  import React, { Component } from "react";
2
3  export const Moviesprovider = React.createContext();
4
5  export default class MoviesContext extends Component {
6    constructor() {
7      super();
8      this.state = {
9        director: "Christopher Nplan",
10       movie: "Intestellar",
11     };
12   }
13   render() {
14     return (
15       <Moviesprovider.Provider value={{ ...this.state }}>
16         {this.props.children}
17       </Moviesprovider.Provider>
18     );
19   }
20 }
21
```

```
JS App.js × JS ChildBcom.js × JS index.js
src > Assig-No-6 > JS ChildBcom.js > ...
1  import React, { useContext } from "react";
2  import { Moviesprovider } from "../MoviesContext";
3
4  function ChildBcom() {
5    const obj = useContext(Moviesprovider);
6    return (
7      <div>
8        <h1 style={{ backgroundColor: "brown" }}>This is Child B component</h1>
9        <h1>{obj.movie}</h1>
10      </div>
11    );
12  }
13
14  export default ChildBcom;
15
```

```
JS App.js JS ChildAcom.js X JS index.js
src > Assig-No-6 > JS ChildAcom.js > ...
1 import React, { useContext } from "react";
2 import { Moviesprovider } from "../MoviesContext";
3
4 function ChildAcom() {
5   const obj = useContext(Moviesprovider);
6   return (
7     <div>
8       <h1 style={{ backgroundColor: "green" }}>This is A Child component</h1>
9       <h1>{obj.director}</h1>
10    </div>
11  );
12 }
13
14 export default ChildAcom;
15
```

Code of the Sol GitHub Link:-

<https://github.com/Jeevan-kumar-Raj/Small-Tasks>

Submitted By: - jeevan kumar