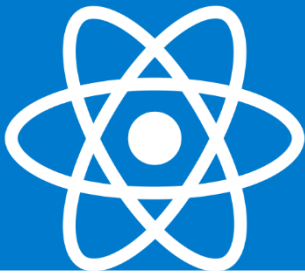




React JS

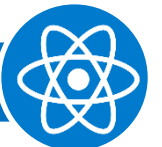
#React Notes

| LocalStorage



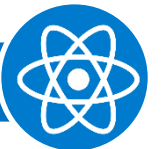
What is LocalStorage?

- LocalStorage is a web storage object to store the data on the user's computer locally, which means the stored data is saved across browser sessions and the data stored has no expiration time.
- LocalStorage data can be accessible to origin only.



What is use of LS?

- Auth
- Users Choice
 - Font Background Color



Methods :-

- **setItem()**

- This method is used to add a key and a value to localStorage.

- **getItem()**

- This method is used to get an item from localStorage using the key.

- **removeItem()**

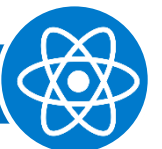
- This technique is used to delete an item from localStorage based on its key.

- **clear()**

- This technique is used to delete all instances of localStorage.

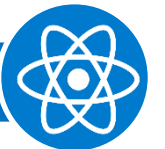
- **key()**

- When you supply a number, it aids in the retrieval of a localStorage key.



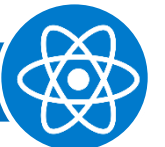
Syntax:-

- To store data
 - `localStorage.setItem('Name', 'Akash');`
- To retrieve data
 - `localStorage.getItem('Name');`
- To clear a specific item
 - `localStorage.removeItem('Name');`
- To clear the whole data stored in localStorage
 - `localStorage.clear();`
- Get n-th key name
 - `localStorage.key(1);`



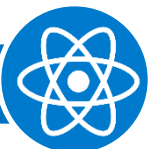
Storing objects in local storage

- If we want to store objects instead of primitive values, like strings or numbers, we run into the issue that instead of the object, we'll find a string in local storage that reads [object Object]
- That's because the toString function of the object prototype returns this. To store objects in local storage, we need to stringify the object first, and then parse it when we try to read from local storage. The JSON API allows us to do this:



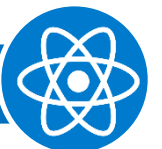
Store Object in LocalStorage

- Step 1: Stringify the object
 - Convert Javascript Object into String
 - `localStorage.setItem('user', JSON.stringify({name: 'akash', age: ' 30' }));`
- Step 2: Parse the object
 - Convert String into Javascript Object
 - `const user = JSON.parse(localStorage.getItem('user'));`

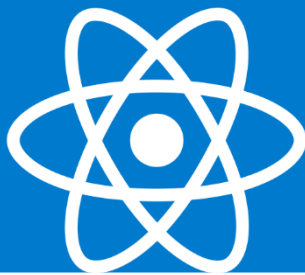


Example 2

- **Javascript Object Store as String using JSON.Stringify**
 - `const person = { firstName: 'Akash', lastName: 'Padhiyar' };`
 - `localStorage.setItem('user', JSON.stringify(person));`
- **Fetch String from LocalStorage and Convert into JavaScript Object**
 - `const stringifiedPerson = localStorage.getItem('user');`
 - `const personAsObjectAgain = JSON.parse(stringifiedPerson);`



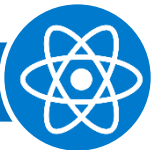
| Class Example



Example

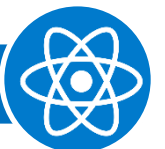
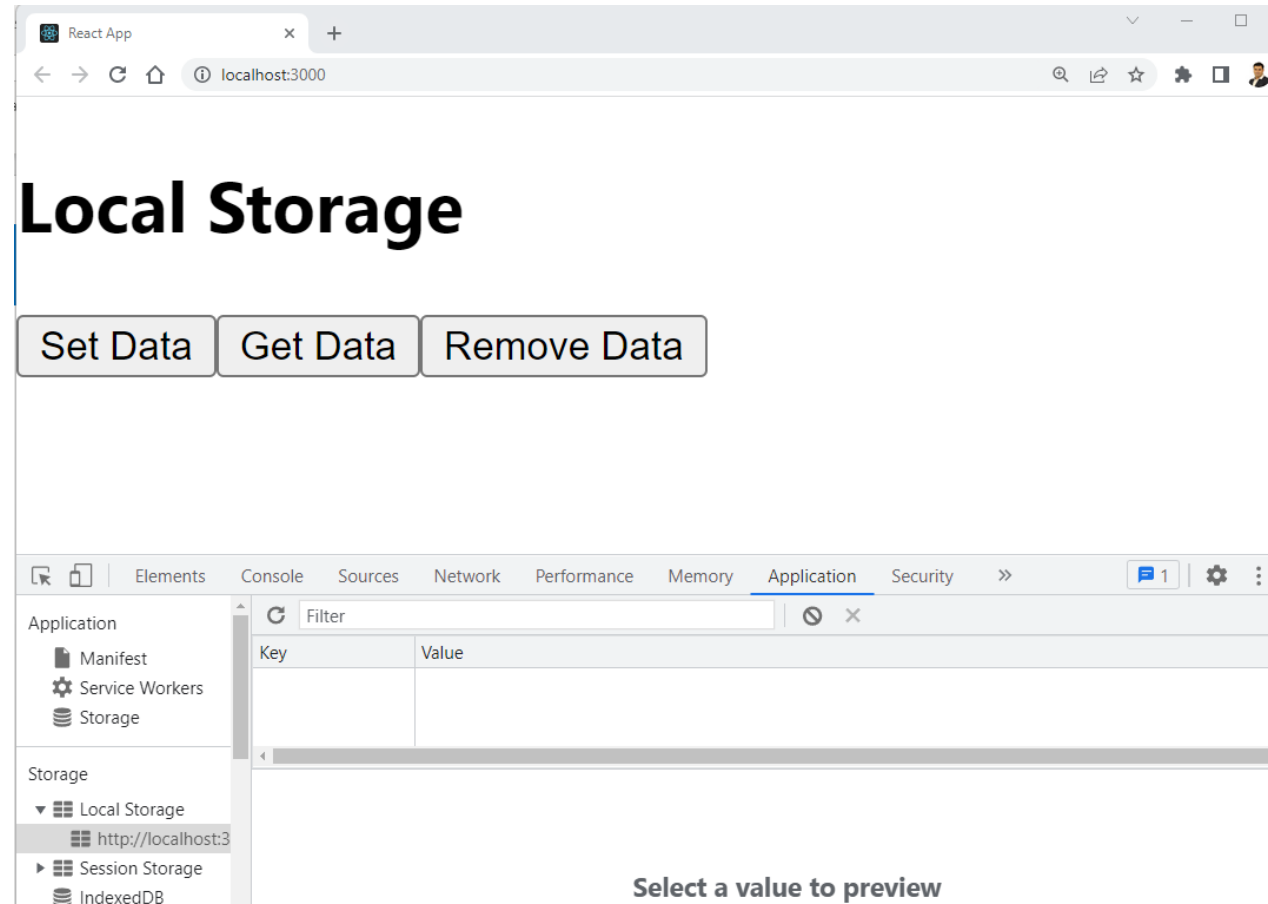
```
App.js
src > App.js > App
1 import React from 'react';
2 class App extends React.Component {
3   constructor(props) {
4     super(props);
5     this.state = {}
6   };
7   //Set Data
8   setData(){
9     localStorage.setItem("mycolor",'red');
10    alert("Data is set");
11  }
12  //Get Data
13  getData(){
14    let mydata = localStorage.getItem("mycolor");
15    if(mydata)
16    {
17      alert("Data is: " + mydata);
18    }else{
19      alert("Data is not set");
20    }
21  }
22  //Remove Data
23  removeData(){
24    localStorage.removeItem('mycolor');
25    alert("Data Removed");
26  }
```

```
27   render() {
28     return (
29       <div>
30         <h2>Local Storage</h2>
31         <button onClick={this.setData.bind(this)}>Set Data</button>
32         <button onClick={this.getData.bind(this)}>Get Data</button>
33         <button onClick={this.removeData.bind(this)}>Remove Data</button>
34       </div>
35     );
36   }
37 }
38
39 export default App;
```

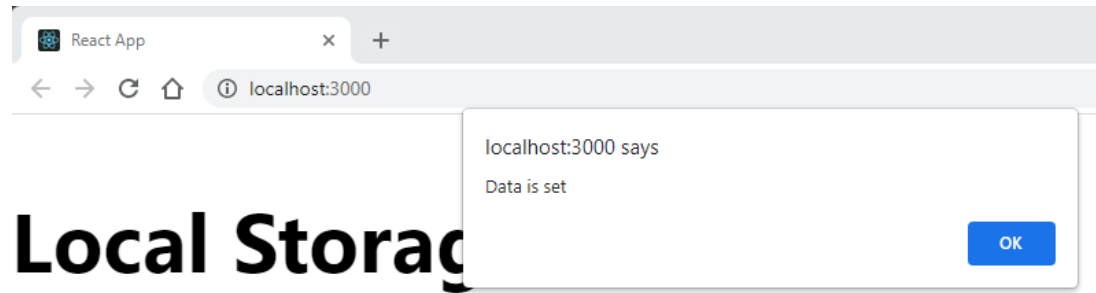


Local Storage

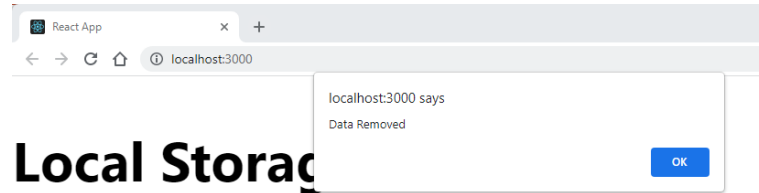
- To View LocalStorage data open inspect->Application->Storage->LocalStorage



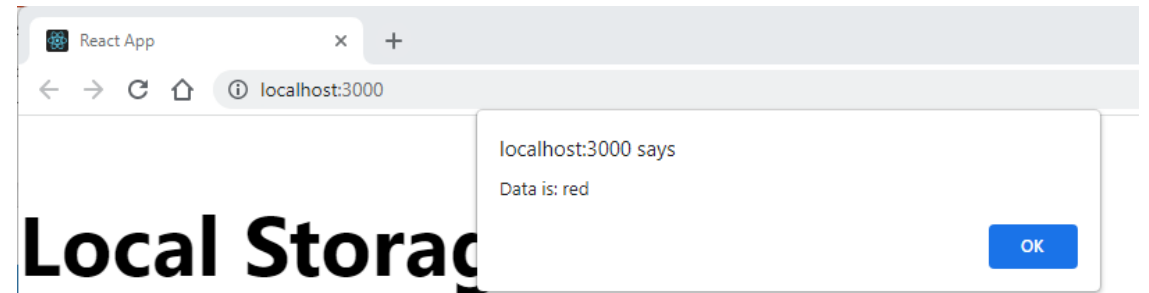
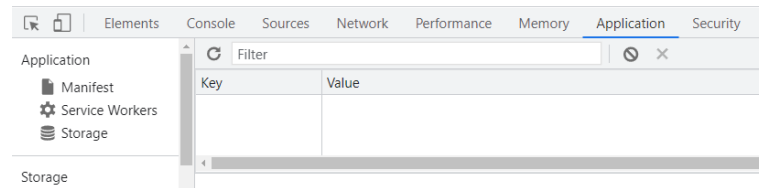
Example : Set Get Remove



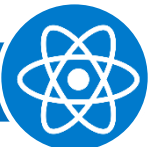
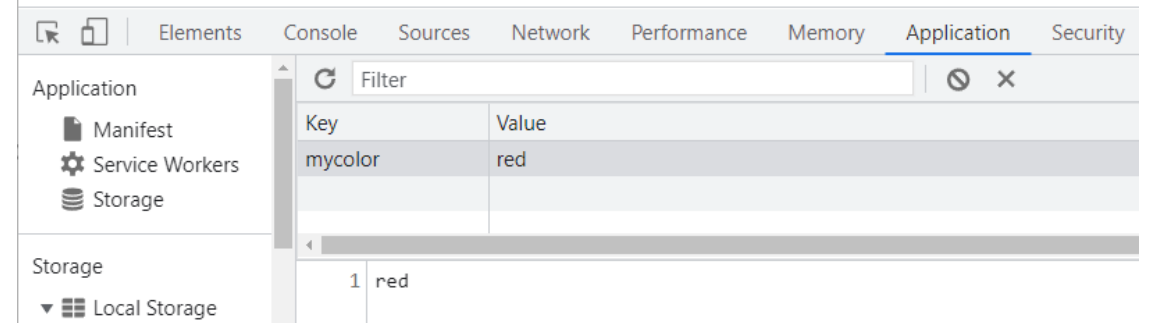
Set Data Get Data Remove Data



Set Data Get Data Remove Data



Set Data Get Data Remove Data

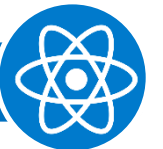


Code

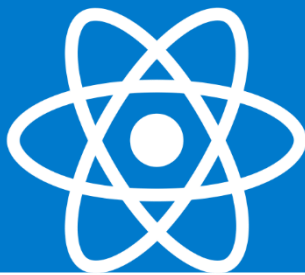
```
import React from 'react';
class App extends React.Component {
  constructor(props) {
    super(props);
    this.state = {}
  };

  //Set Data
  setData(){
    localStorage.setItem("mycolor",'red');
    alert("Data is set");
  }
  //Get Data
  getData(){
    let mydata = localStorage.getItem("mycolor");
    if(mydata)
    {
      alert("Data is: " + mydata);
    }else{
      alert("Data is not set");
    }
  }
  //Remove Data
  removeData(){
    localStorage.removeItem('mycolor');
    alert("Data Removed");
  }
}
```

```
render() {
  return (
    <div>
      <h2>Local Storage</h2>
      <br/>
      <button onClick={this.setData.bind(this)}>Set Data</button>
      <button onClick={this.getData.bind(this)}>Get Data</button>
      <button onClick={this.removeData.bind(this)}>Remove Data</button>
    </div>
  );
}
export default App;
```



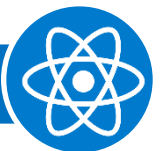
| Local Storage with StateManagement



LocalStorage with State /SetState

■ On Load Print Local Storage Value

```
src > js App.js > App
1  import React from 'react';
2  class App extends React.Component {
3      constructor(props) {
4          super(props);
5          this.state = {
6              txt1: '',
7          }
8      };
9      //On Load Check Data
10     componentDidMount() {
11         let mydata = localStorage.getItem("mycolor");
12         if(mydata)
13         {
14             this.setState({txt1: mydata});
15         }else{
16             this.setState({txt1: "Data is not set"});
17         }
18     }
19
20     //Set Data
21     setData(){
22         var data = "red";
23         localStorage.setItem("mycolor", 'red');
24         this.setState({txt1: data});
25         alert("Data is set");
26     }
27
28     //Get Data
29     getData(){
30         let mydata = localStorage.getItem("mycolor");
31         if(mydata)
32         {
33             this.setState({txt1: mydata});
34             alert("Data is: " + mydata);
35         }else{
36             this.setState({txt1: "Data is not set"});
37             alert("Data is not set");
38         }
39     }
40
41     //Remove Data
42     removeData(){
43         localStorage.removeItem('mycolor');
44         this.setState({txt1: "Data Removed"});
45         alert("Data Removed");
46     }
47
48     render() {
49         return (
50             <div>
51                 <h2>Local Storage</h2>
52                 Value is : {this.state.txt1}
53                 <br/>
54                 <button onClick={this.setData.bind(this)}>Set Data</button>
55                 <button onClick={this.getData.bind(this)}>Get Data</button>
56                 <button onClick={this.removeData.bind(this)}>Remove Data</button>
57             </div>
58         );
59     }
60 }
61 export default App;
```



Output

← → ↻ 🏠 ⓘ localhost:3000

Local Storage

Value is : Data is not set

Set Data

Get Data

Remove Data

← → ↻ 🏠 ⓘ localhost:3000

Local Storage

Value is : red

Set Data

Get Data

Remove Data

← → ↻ 🏠 ⓘ localhost:3000

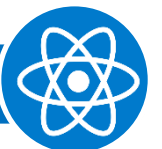
Local Storage

Value is : Data Removed

Set Data

Get Data

Remove Data



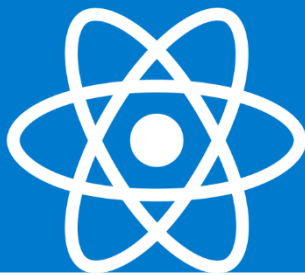
Code

```
import React from 'react';
class App extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      txt1: "",
    }
  };
  //On Load Check Data
  componentDidMount() {
    let mydata = localStorage.getItem("mycolor");
    if(mydata)
    {
      this.setState({txt1: mydata});
    }else{
      this.setState({txt1: "Data is not set"});
    }
  }
  //Set Data
  setData(){
    var data = "red";
    localStorage.setItem("mycolor",'red');
    this.setState({txt1: data});
    alert("Data is set");
  }
}
```

```
//Get Data
getData(){
  let mydata = localStorage.getItem("mycolor");
  if(mydata)
  {
    this.setState({txt1: mydata});
    alert("Data is: " + mydata);
  }else{
    this.setState({txt1: "Data is not set"});
    alert("Data is not set");
  }
}
//Remove Data
removeData(){
  localStorage.removeItem('mycolor');
  this.setState({txt1: "Data Removed"});
  alert("Data Removed");
}
render() {
  return (
    <div>
      <h2>Local Storage</h2>
      Value is : {this.state.txt1}
      <br/>
      <button onClick={this.setData.bind(this)}>Set Data</button>
      <button onClick={this.getData.bind(this)}>Get Data</button>
      <button onClick={this.removeData.bind(this)}>Remove Data</button>
    </div>
  );
}
}
export default App;
```



| Object



Save Remove Code

```
Js App.js M X
src > Js App.js > App
1 import React from 'react'
2 class App extends React.Component {
3   constructor(props) {
4     super(props);
5     this.state = {  };
6   }
7   saveData(){
8     //Object
9     var myuser = {name:"akash",gender:"male",age:30}
10    //Object as String
11    var mydata = JSON.stringify(myuser);
12    //Store Object as String
13    localStorage.setItem('userdata', mydata)
14    this.setState({msg:"Saved"});
15  }
16  removeData(){
17    localStorage.removeItem('userdata');
18    this.setState({msg:"Removed"});
19    this.setState({name:'.name,gender:'.gender,age:'.age});
20  }
21 }
```

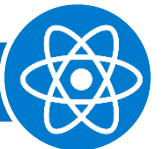
```
import React from 'react'
class App extends React.Component {
  constructor(props) {
    super(props);
    this.state = {  };
  }
  saveData(){
    //Object
    var myuser = {name:"akash",gender:"male",age:30}
    //Object as String
    var mydata = JSON.stringify(myuser);
    //Store Object as String
    localStorage.setItem('userdata', mydata)
    this.setState({msg:"Saved"});
  }
  removeData(){
    localStorage.removeItem('userdata');
    this.setState({msg:"Removed"});
    this.setState({name:".name,gender:".gender,age:"});
  }
}
```



Get Code

```
Js App.js M X
src > Js App.js > App > render
23   getData() {
24     //Convert to Object
25     if (localStorage.getItem('userdata')) {
26       var mydata = JSON.parse(localStorage.getItem('userdata'));
27       console.log(mydata);
28       //Assign Object Value to State
29       this.setState({ name: mydata.name, gender: mydata.gender, age: mydata.age });
30     } else {
31       this.setState({ msg: "No Value" });
32     }
33   }
34   render() {
35     return (
36       <div>
37         <h2>Local Storage</h2>
38         {this.state.msg}<br />
39         Name : {this.state.name}<br />
40         Gender : {this.state.gender}<br />
41         Age : {this.state.age}
42         <br />
43         <button onClick={this.saveData.bind(this)}>Save</button>
44         <button onClick={this.getData.bind(this)}>Get</button>
45         <button onClick={this.removeData.bind(this)}>Remove</button>
46       </div>
47     );
48   }
49 }
50 export default App;
```

```
getData() {
  //Convert to Object
  if (localStorage.getItem("userdata")) {
    var mydata = JSON.parse(localStorage.getItem('userdata'));
    console.log(mydata);
    //Assign Object Value to State
    this.setState({ name: mydata.name, gender: mydata.gender, age: mydata.age });
  } else {
    this.setState({ msg: "No Value" });
  }
}
render() {
  return (
    <div>
      <h2>Local Storage</h2>
      {this.state.msg}<br />
      Name : {this.state.name}<br />
      Gender : {this.state.gender}<br />
      Age : {this.state.age}
      <br />
      <button onClick={this.saveData.bind(this)}>Save</button>
      <button onClick={this.getData.bind(this)}>Get</button>
      <button onClick={this.removeData.bind(this)}>Remove</button>
    </div>
  );
}
export default App;
```

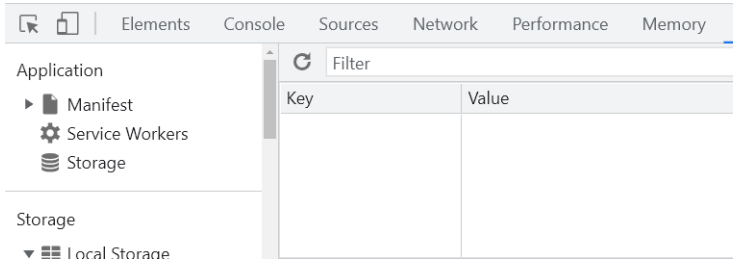


Output

Local Storage

Name :
Gender :
Age :

Save Get Remove



Application

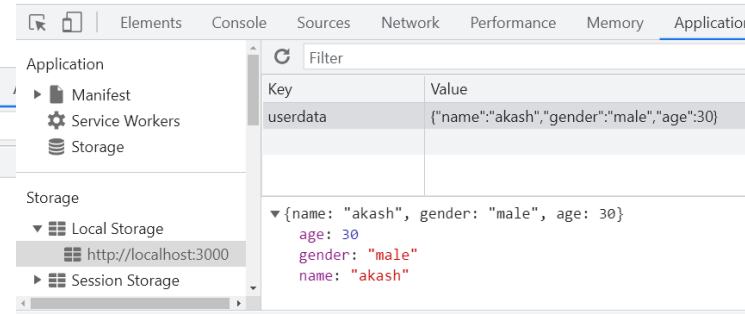
- Manifest
- Service Workers
- Storage
 - Local Storage
 - http://localhost:3000
 - Session Storage

Key	Value
-----	-------

Local Storage

Saved
Name :
Gender :
Age :

Save Get Remove



Application

- Manifest
- Service Workers
- Storage
 - Local Storage
 - http://localhost:3000
 - Session Storage

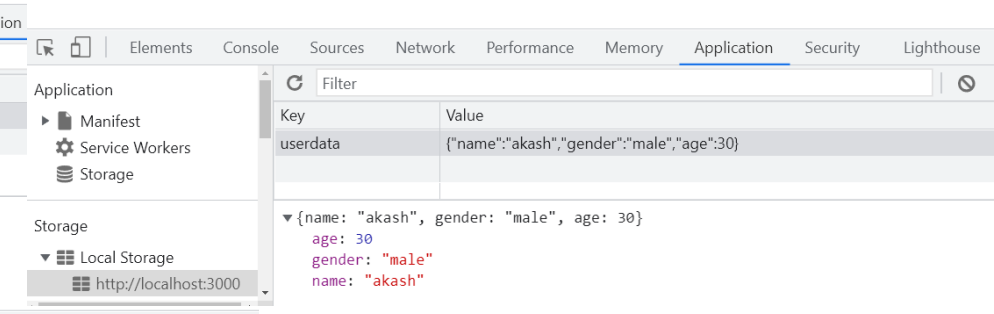
Key	Value
userdata	{"name":"akash","gender":"male","age":30}

```
{name: "akash", gender: "male", age: 30}
age: 30
gender: "male"
name: "akash"
```

Local Storage

Saved
Name : akash
Gender : male
Age : 30

Save Get Remove

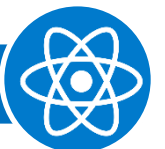


Application

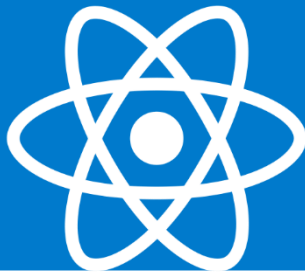
- Manifest
- Service Workers
- Storage
 - Local Storage
 - http://localhost:3000

Key	Value
userdata	{"name":"akash","gender":"male","age":30}

```
{name: "akash", gender: "male", age: 30}
age: 30
gender: "male"
name: "akash"
```



| Function

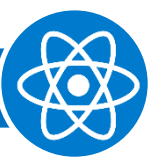


Set Get Remove (Function)

JS App.js M X

src > JS App.js > App

```
1  import React from 'react'
2  function App() {
3
4      const [value, setValue] = React.useState('LocalStorage');
5
6      const setData = () => {
7          setValue('Value is Set');
8          localStorage.setItem('name', "Akash");
9      }
10
11     const getData = () => {
12         var a = localStorage.getItem('name');
13         setValue('Value is : ' + a);
14     }
15
16     const removeData = () => {
17         localStorage.removeItem('name');
18         setValue('Value is Removed');
19     }
20
21     return (
22         <div>
23             { value }
24             <br />
25             <button onClick={setData}>Set</button>
26             <button onClick={getData}>Get</button>
27             <button onClick={removeData}>Remove</button>
28         </div>
29     );
30 }
31 export default App;
```



React App

localhost:3001

LocalStorage

SetGetRemove

React App

localhost:3001

Value is Set

SetGetRemove

React App

localhost:3001

Value is : Akash

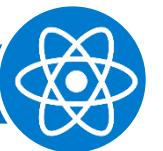
SetGetRemove

React App

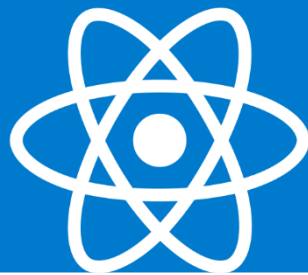
localhost:3001

Value is Removed

SetGetRemove



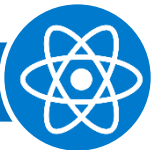
| Counter with LocalStorage



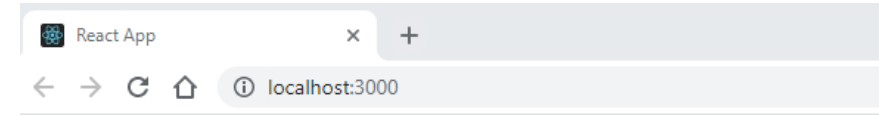
Counter with LocalStorage

```
1 import React from 'react';
2 class App extends React.Component {
3   constructor(props) {
4     super(props);
5     this.state = {}
6   };
7   //On Load Check Data
8   componentDidMount() {
9     let mydata = localStorage.getItem("counter");
10    if(mydata)
11    {
12      this.setState({txt1: parseInt(mydata)});
13    }else{
14      this.setState({txt1:0});
15    }
16  }
17  //Set Data
18  incrementData(){
19    let mydata = localStorage.getItem("counter");
20    if(mydata)
21    {
22      var no1 = parseInt(mydata) + 1;
23      this.setState({txt1:no1});
24      localStorage.setItem("counter", no1);
25    }else{
26      localStorage.setItem("counter", 0);
27      this.setState({txt1:0});
28    }
29  }
```

```
30 //Get Data
31 decrementData(){
32   let mydata = localStorage.getItem("counter");
33   if(mydata)
34   {
35     var no1 = parseInt(mydata) - 1;
36     this.setState({txt1:no1});
37     localStorage.setItem("counter", no1);
38   }else{
39     localStorage.setItem("counter", 0);
40     this.setState({txt1:0});
41   }
42 }
43 //Remove Data
44 removeData(){
45   localStorage.removeItem('counter');
46   this.setState({txt1: "Data Removed"});
47 }
48 render() {
49   return (
50     <div>
51       <h2>Local Storage</h2>
52       Value is : {this.state.txt1}
53       <br/>
54       <button onClick={this.incrementData.bind(this)}>+ Data</button>
55       <button onClick={this.decrementData.bind(this)}>- Data</button>
56       <button onClick={this.removeData.bind(this)}>X Data</button>
57     </div>
58   );
59 }
60 }
61 export default App;
```

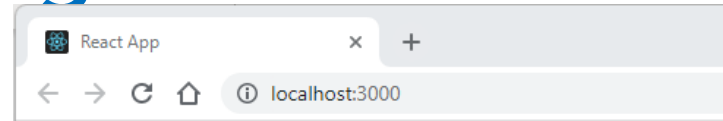
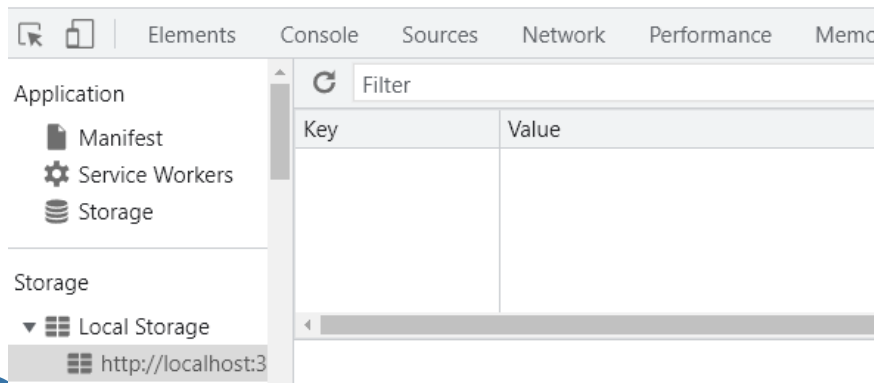


Output LocalStorage



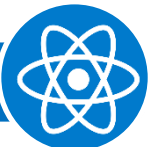
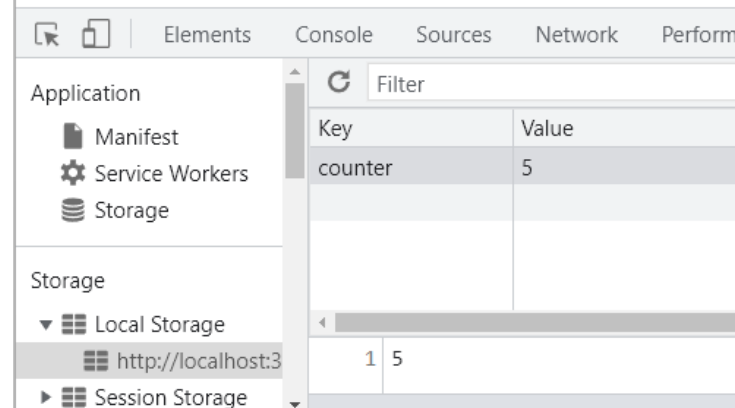
Local Storage

Value is : 0



Local Storage

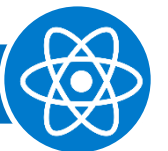
Value is : 5



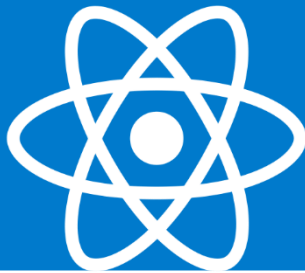
Code

```
import React from 'react';
class App extends React.Component {
  constructor(props) {
    super(props);
    this.state = {}
  };
  //On Load Check Data
  componentDidMount() {
    let mydata = localStorage.getItem("counter");
    if(mydata)
    {
      this.setState({txt1: parseInt(mydata)});
    }else{
      this.setState({txt1:0});
    }
  }
  //Set Data
  incrementData(){
    let mydata = localStorage.getItem("counter");
    if(mydata)
    {
      var no1 = parseInt(mydata) + 1;
      this.setState({txt1:no1});
      localStorage.setItem("counter", no1);
    }else{
      localStorage.setItem("counter", 0);
      this.setState({txt1:0});
    }
  }
}
```

```
//Get Data
decrementData(){
  let mydata = localStorage.getItem("counter");
  if(mydata)
  {
    var no1 = parseInt(mydata) - 1;
    this.setState({txt1:no1});
    localStorage.setItem("counter", no1);
  }else{
    localStorage.setItem("counter", 0);
    this.setState({txt1:0});
  }
}
//Remove Data
removeData(){
  localStorage.removeItem('counter');
  this.setState({txt1: "Data Removed"});
}
render() {
  return (
    <div>
      <h2>Local Storage</h2>
      Value is : {this.state.txt1}
      <br/>
      <button onClick={this.incrementData.bind(this)}>+ Data</button>
      <button onClick={this.decrementData.bind(this)}>- Data</button>
      <button onClick={this.removeData.bind(this)}>X Data</button>
    </div>
  );
}
export default App;
```



| Function



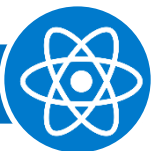
Example : Save Get Delete

```
App.js
src > App.js > App > deleteData
1  import { React } from "react";
2
3  function App() {
4    //Save Data
5    let saveData = () =>{
6      localStorage.setItem('mycolor', "red");
7      alert('Record Added')
8    }
9    //Get Data
10   let getData = () =>{
11     var data = localStorage.getItem("mycolor")
12     alert("Value is " + data)
13   }
14   //Get Data
15   let deleteData = () =>{
16     localStorage.removeItem("mycolor")
17     alert("Record Deleted")
18   }
19   return (
20     <div>
21       <h2>LocalStorage</h2>
22       <button onClick={saveData}>Save Data</button>
23       <button onClick={getData}>Display Data</button>
24       <button onClick={deleteData}>Delete Data </button>
25     </div>
26   );
27 }
```

```
import { React } from "react";

function App() {
  //Save Data
  let saveData = () =>{
    localStorage.setItem('mycolor', "red");
    alert('Record Added')
  }
  //Get Data
  let getData = () =>{
    var data = localStorage.getItem("mycolor")
    alert("Value is " + data)
  }
  //Get Data
  let deleteData = () =>{
    localStorage.removeItem("mycolor")
    alert("Record Deleted")
  }
  return (
    <div>
      <h2>LocalStorage</h2>
      <button onClick={saveData}>Save Data</button>
      <button onClick={getData}>Display Data</button>
      <button onClick={deleteData}>Delete Data </button>
    </div>
  );
}

export default App;
```



Toggle Content Show

```
src > App.js > default
1  import * as React from 'react';
2
3  const App = () => {
4    const [isOpen, setOpen] = React.useState(
5      JSON.parse(localStorage.getItem('is-open')) || false
6    );
7
8    const handleToggle = () => {
9      setOpen(!isOpen);
10   };
11
12   React.useEffect(() => {
13     localStorage.setItem('is-open', JSON.stringify(isOpen));
14   }, [isOpen]);
15
16   return (
17     <div>
18       <button onClick={handleToggle}>Toggle</button>
19       {isOpen && <div>Content</div>}
20     </div>
21   );
22 };
23
24 export default App;
```

```
import * as React from 'react';
```

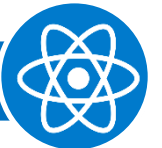
```
const App = () => {
  const [isOpen, setOpen] = React.useState(
    JSON.parse(localStorage.getItem('is-open')) || false
  );
```

```
  const handleToggle = () => {
    setOpen(!isOpen);
  };
```

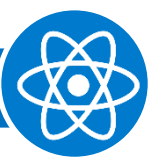
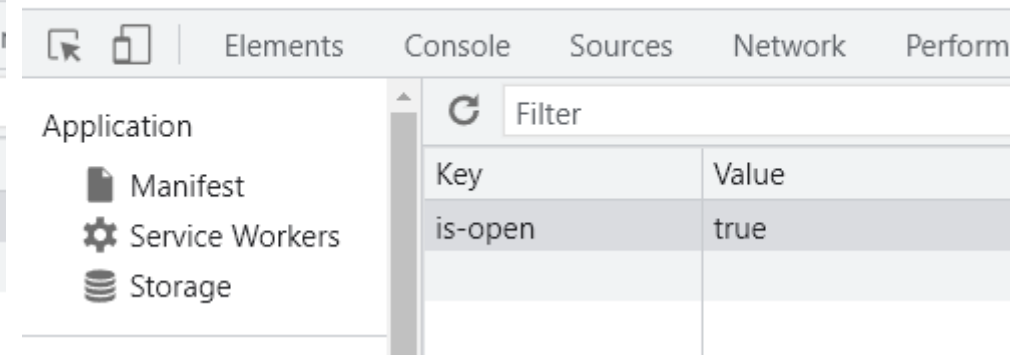
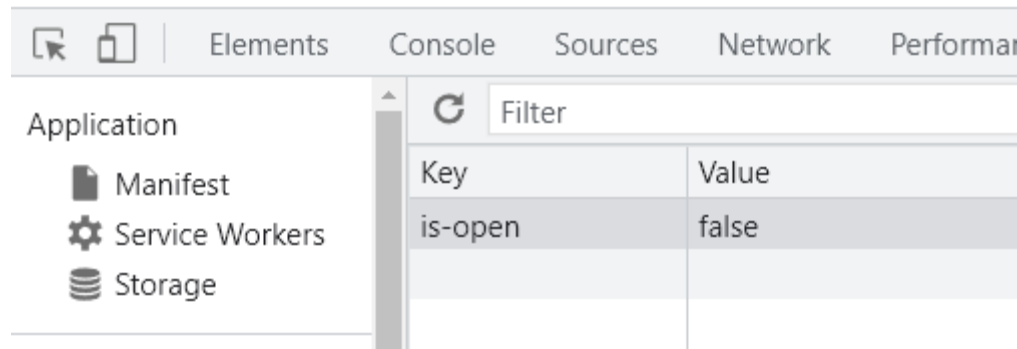
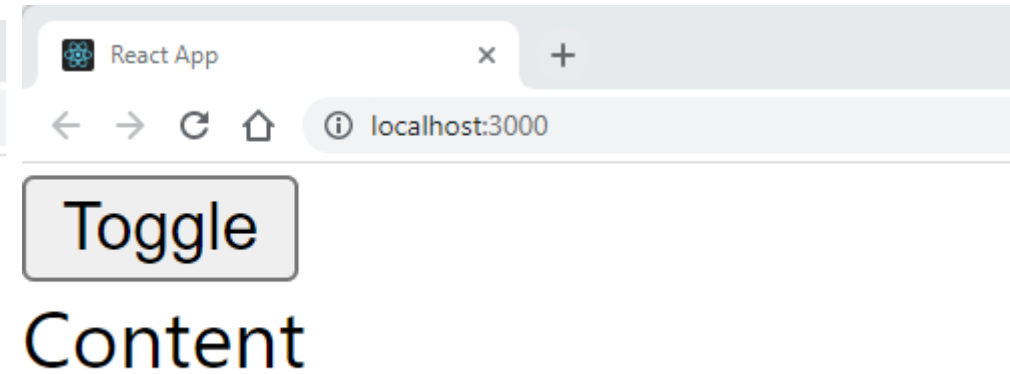
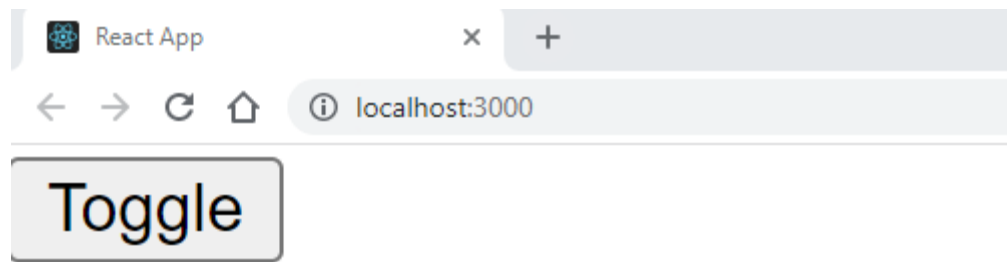
```
  React.useEffect(() => {
    localStorage.setItem('is-open', JSON.stringify(isOpen));
  }, [isOpen]);
```

```
  return (
    <div>
      <button onClick={handleToggle}>Toggle</button>
      {isOpen && <div>Content</div>}
    </div>
  );
};
```

```
export default App;
```



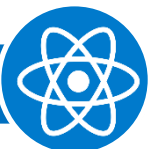
Toggle Content



Color Change

```
src > App.js > App
1  import { useState } from "react";
2
3  function App() {
4
5    const [bgColor, setBgColor] = useState("");
6
7    const handleClick = (color) => {
8      localStorage.setItem("color", color);
9      setBgColor(color);
10   };
11
12   return (
13     <div
14       style={{ backgroundColor: localStorage.getItem("color"), height: 500 }}
15     >
16       <button onClick={() => handleClick("red")}>Red</button>
17       <button onClick={() => handleClick("blue")}>Blue</button>
18       <button onClick={() => handleClick("Yellow")}>Yellow</button>
19     </div>
20   );
21 }
22 export default App;
```

```
import { useState } from "react";
function App() {
  const [bgColor, setBgColor] = useState("");
  const handleClick = (color) => {
    localStorage.setItem("color", color);
    setBgColor(color);
  };
  return (
    <div
      style={{ backgroundColor: localStorage.getItem("color"), height: 500 }}
    >
      <button onClick={() => handleClick("red")}>Red</button>
      <button onClick={() => handleClick("blue")}>Blue</button>
      <button onClick={() => handleClick("Yellow")}>Yellow</button>
    </div>
  );
}
export default App;
```





React App

localhost:3000

Red Blue Yellow

Elements Console Sources Network Performance

Application

- Manifest
- Service Workers
- Storage

Key	Value
color	Yellow

React App

localhost:3000

Red Blue Yellow

Elements Console Sources Network Performance

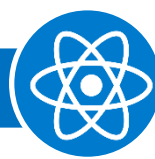
Application

- Manifest
- Service Workers
- Storage

Storage

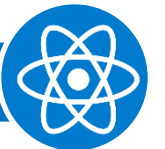
- Local Storage
 - http://localhost:3

Key	Value
color	Yellow



Task

- Create Signup Form having below fields :
 - Name,Email,Password,Mobile (Textbox) and Button.
 - On Button click save all records in LocalStorage (using separate variable or JS Object)
- Create Login Form having below fields.
 - Email and Password (Textbox) and button
 - On Button Click Compare Textbox Email with Localstorage Email and TextBox Password with Local Storage Password.
 - If Value is Same then print Welcome Message on Screen else Error Message



Page 1 Signup

Name

Email

Password

Mobile

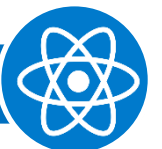
Store all Textbox Value in
Local Storage in Seprate
Variable

Page 2 Login

Email

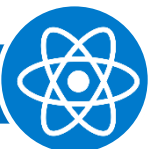
Password

OnClick Check
Textbox Email & Password == LocalStorage
Email & Password is same or not ?
print respective error or success msg



Part 2

- If Login details are same then redirect on homepage.
- Create New Homepage and Print Logged in Username on Screen.
- Create Change Password Component having Below Fields.
 - Old Password, New Password and Confirm Password Textbox.
 - OnButton Click Compare Old Password is Same as LocalStorage if it is same then Change Password with New Password in LocalStorage .
- Create Logout Button in Homepage on Logout button Click Remove LocalStorage Details and redirect on Login Page.

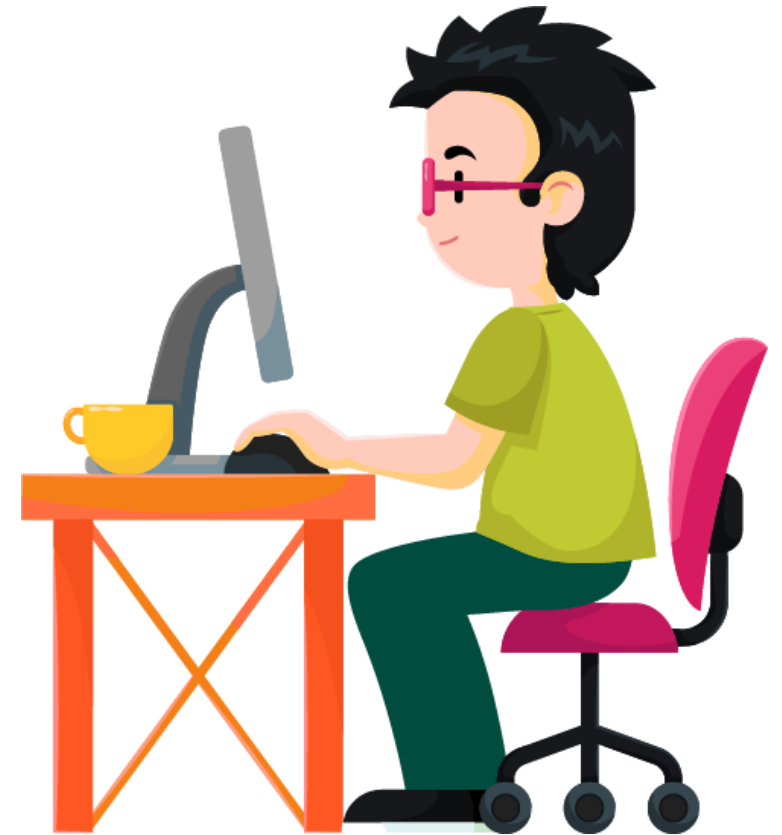


Get Exclusive Video Tutorials



www.apptutorials.com

<https://www.youtube.com/user/Akashtips>





Get More Details

www.akashsir.com



If You Liked It !

Rating Us Now



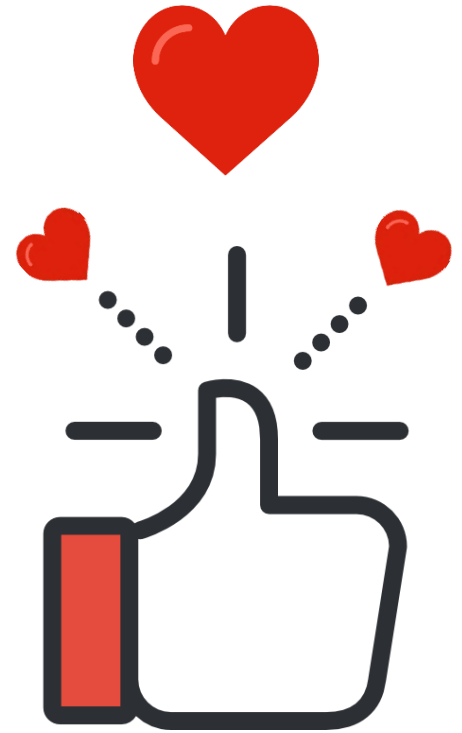
Just Dial

https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4_BZDET



Sulekha

<https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad>



Connect With Me



Akash Padhiyar
#AkashSir

www.akashsir.com

www.akashtechlabs.com

www.akashpadhiyar.com

www.apptutorials.com

Social Info



Akash.padhiyar



Akashpadhiyar



Akash_padhiyar



+91 99786-21654



#Akashpadhiyar

#apptutorials