JS

JavaScript Class

#JavaScript Notes

Storage

What is the Web Storage API?

- The Web Storage API is a set of mechanisms that enable browsers to store key-value pairs. It is designed to be much more intuitive than using cookies.

What is localStorage in JavaScript?

- The localStorage is an instance of the Storage type that allows you to store persistent data in the web browsers.
- localStorage is a property that allows JavaScript sites and apps to save key-value pairs in a web browser with no expiration date.
- This means the data stored in the browser will persist even after the browser window is closed / System Restart .
- The storage is bound to the origin (domain/protocol/port triplet).
- That is, different protocols or subdomains infer different storage objects, they can’t access data from each other.
- The localStorage can store only strings. To store objects, you convert them to strings using the JSON.stringify() method.
- And you convert the strings into objects when you retrieve them from the localStorage using the JSON.parse() method

localStorage vs. cookies

- First, the data stored in the localStorage isn’t sent to the server in every request like cookies. For this reason, you can store more data in the localStorage.

- Most modern web browsers allow you to store up to 5MB of data in the localStorage. Note that you can store up to 4KB in cookies.

- Second, the data stored in the localStorage can be managed by the client, specifically JavaScript in the web browser. It cannot be accessible by the servers.

- However, cookies can be managed by both JavaScript in web browsers and servers.

localStorage limitations

- Do not store sensitive user information in localStorage
- It is not a substitute for a server based database as information is only stored on the browser
- localStorage is limited to 5MB across all major browsers
- localStorage is quite insecure as it has no form of data protection and can be accessed by any code on your web page
- localStorage is synchronous, meaning each operation called would only execute one after the other

What is Use ?

- User Auth
- Users Details
  - Font Size
  - Font Color

Where is localStorage stored?

- In Google Chrome, web storage data is saved in an SQLite file in a subfolder in the user's profile.
- The subfolder is located at \AppData\Local\Google\Chrome\User Data\Default\Local Storage on Windows machines and
- ~/Library/Application Support/Google/Chrome/Default/Local Storage on macOS

- Firefox saves storage objects in an SQLite file called webappsstore.sqlite, which is also located in the user's profile folder.

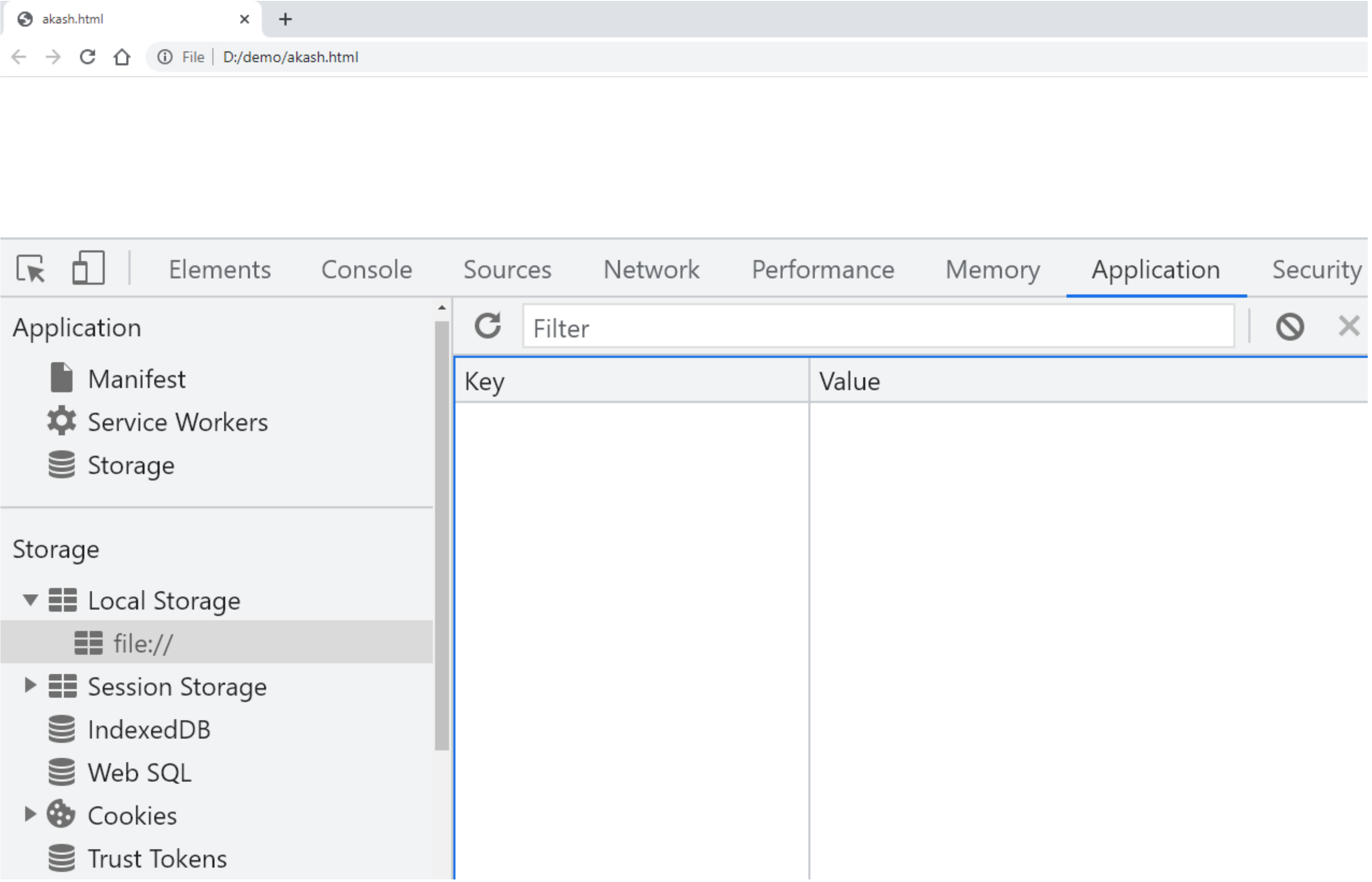
How does localStorage work?

- To use localStorage in your web applications, there are five methods to choose from:

- **setItem():** Add key and value to localStorage
- **getItem():** This is how you get items from localStorage
- **removeItem():** Remove an item by key from localStorage
- **clear():** Clear all localStorage
- **key():** Passed a number to retrieve the key of a localStorage

- Note :Since the window object is global, you don't need to explicitly specify it.

Local Storage Option in Chrome



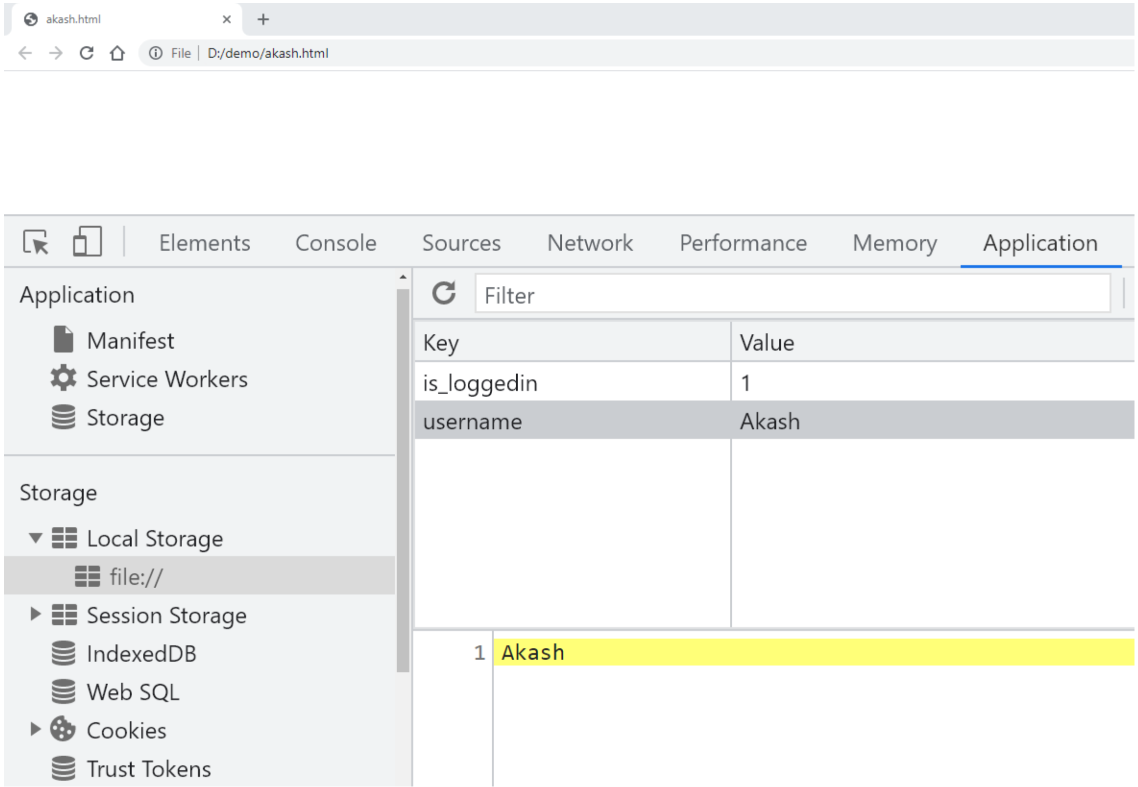
localStorage browser support

- localStorage as a type of web storage is an HTML5 specification. It is supported by major browsers including IE8.
- To be sure the browser supports localStorage, you can check using the following snippet:

```
if (typeof(Storage) !== "undefined") {
  // Code for localStorage
    console.log('Support');
  } else {
  // No web storage Support.
    console.log('No Support');
}
```

How to store values in localStorage

- setItem() method allows you to store values in the localStorage object.
- It takes two parameters: a key and a value. The key can be referenced later to fetch the value attached to it.
- localStorage can only store strings.



D:/demo/akash.html

Key	Value
is_loggedin	1
username	Akash

1 Akash

- Example
  - localStorage.setItem('is_loggedin', 1);
  - localStorage.setItem('username','Akash');

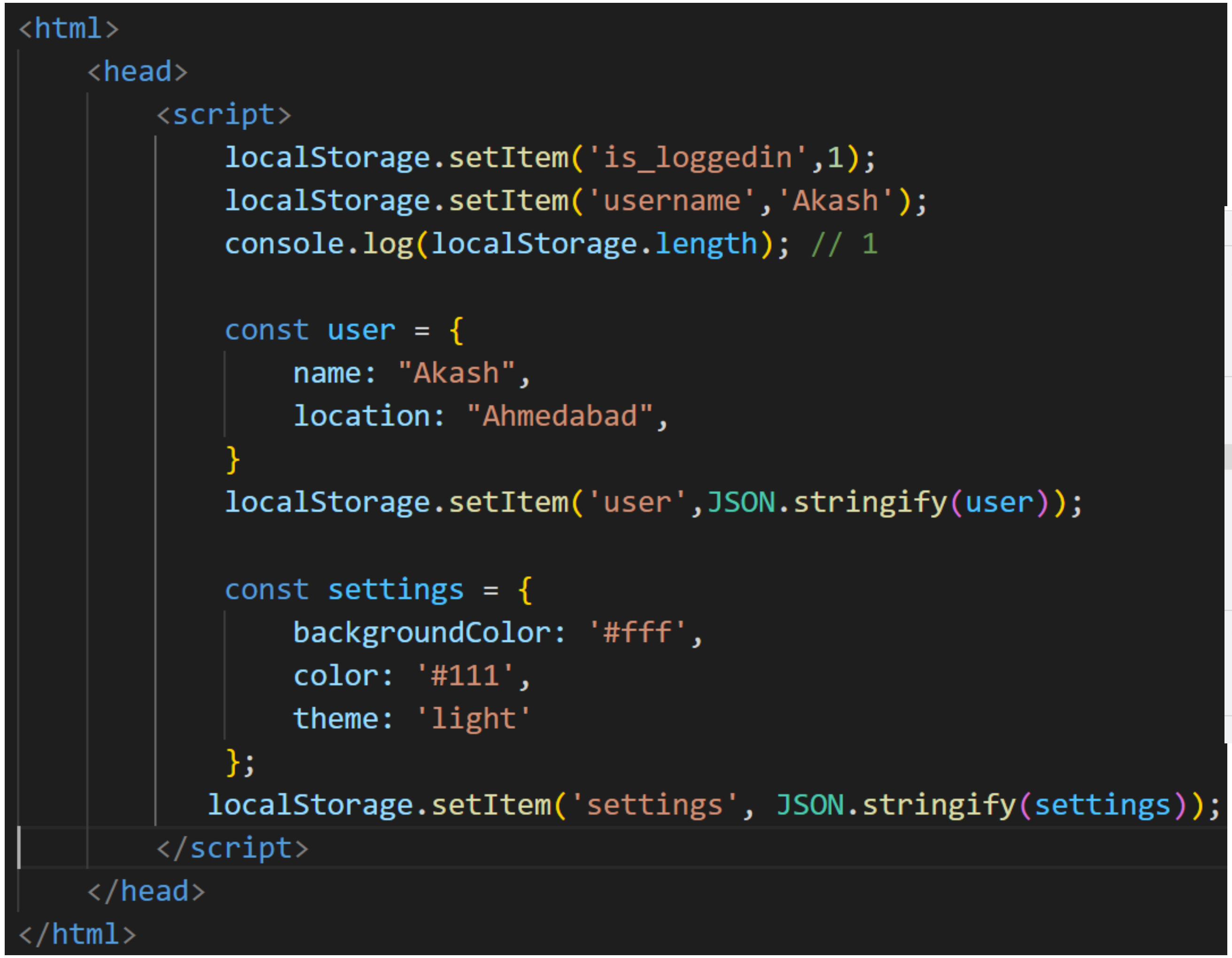
Store Object

- To store arrays or objects, you would have to convert them to strings.
- To do this, we use the JSON.stringify() method before passing to setItem().

```
const user = {
            name: "Akash",
            location: "Ahmedabad",
         }

localStorage.setItem('user',JSON.stringify(user));
```

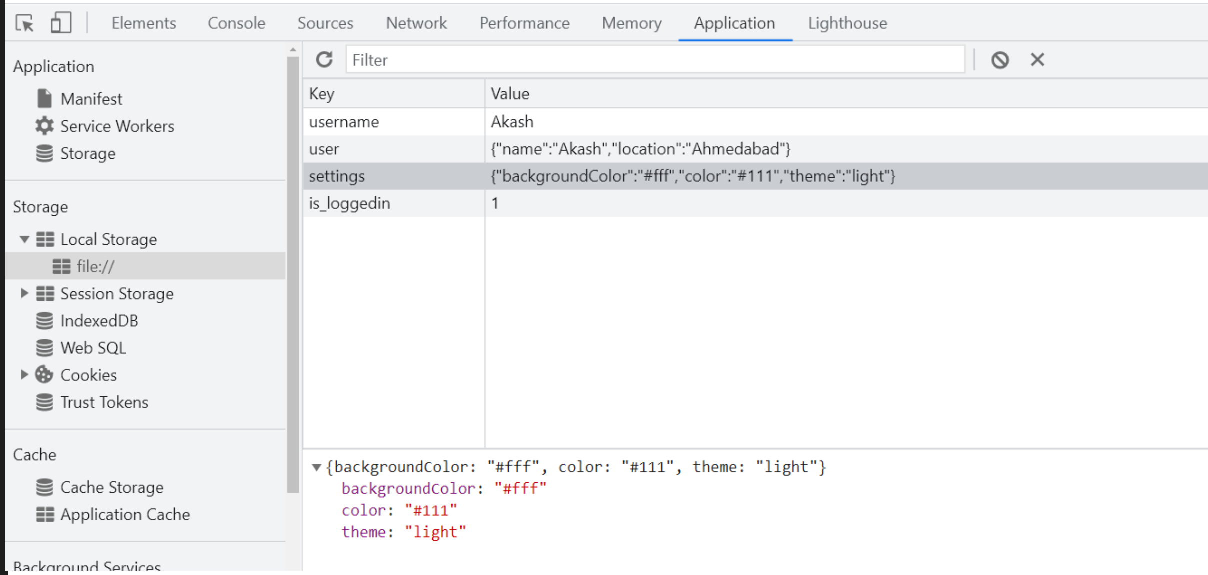
Set Example



```
<html>
    <head>
        <script>
            localStorage.setItem('is_loggedin',1);
            localStorage.setItem('username','Akash');
            console.log(localStorage.length); // 1

            const user = {
                name: "Akash",
                location: "Ahmedabad",
            }
            localStorage.setItem('user',JSON.stringify(user));

            const settings = {
                backgroundColor: '#fff',
                color: '#111',
                theme: 'light'
            };
           localStorage.setItem('settings', JSON.stringify(settings));
        </script>
    </head>
</html>
```



Key	Value
username	Akash
user	{"name":"Akash","location":"Ahmedabad"}
settings	{"backgroundColor":"#fff","color":"#111","theme":"light"}
is_loggedin	1

```
▾{backgroundColor: "#fff", color: "#111", theme: "light"}
   backgroundColor: "#fff"
   color: "#111"
   theme: "light"
```

length property

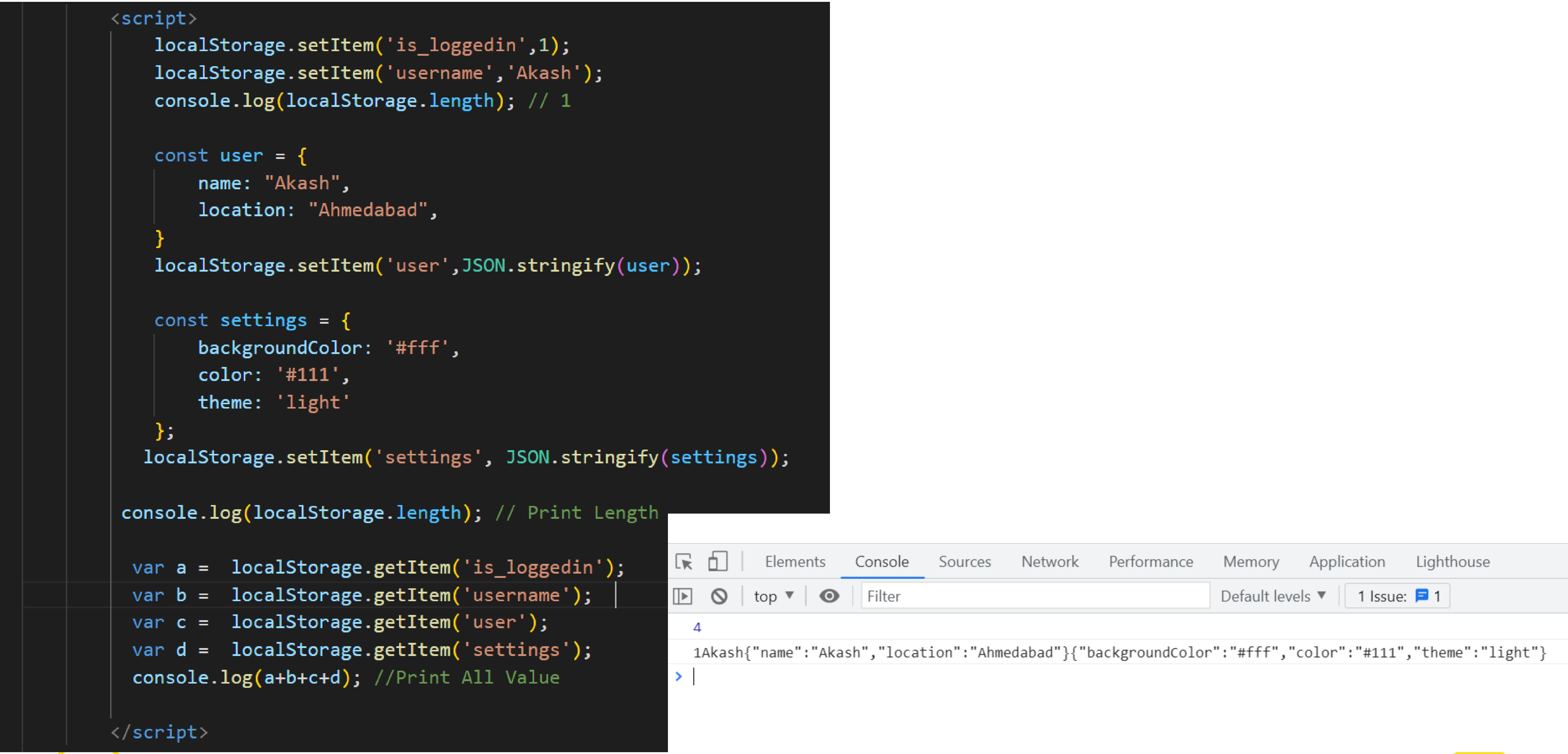
- To get the number of name-value pairs, you use the length property like this:

```
console.log(localStorage.length); // 1
```

How to get items from localStorage

- To get items from localStorage, use the getItem() method. getItem() allows you to access the data stored in the browser’s localStorage object.
- getItem() accepts only one parameter, which is the key, and returns the value as a string.

```
var a =  localStorage.getItem('is_loggedin');
 var b =  localStorage.getItem('username');
 var c =  localStorage.getItem('user');
 var d =  localStorage.getItem('settings');
 console.log(a+b+c+d); //Print All Value
```



```
<script>
    localStorage.setItem('is_loggedin',1);
    localStorage.setItem('username','Akash');
    console.log(localStorage.length); // 1

    const user = {
        name: "Akash",
        location: "Ahmedabad",
    }
    localStorage.setItem('user',JSON.stringify(user));

    const settings = {
        backgroundColor: '#fff',
        color: '#111',
        theme: 'light'
    };
   localStorage.setItem('settings', JSON.stringify(settings));

 console.log(localStorage.length); // Print Length

  var a =  localStorage.getItem('is_loggedin');
  var b =  localStorage.getItem('username');
  var c =  localStorage.getItem('user');
  var d =  localStorage.getItem('settings');
  console.log(a+b+c+d); //Print All Value

</script>
```

```
4
1Akash{"name":"Akash","location":"Ahmedabad"}{"backgroundColor":"#fff","color":"#111","theme":"light"}
>
```

getObject

- localStorage.getItem('settings'); return
- {"backgroundColor":"#fff","color":"#111","theme":"light"}

- To use this value, you would have to convert it back to an object.
- To do this, we make use of the JSON.parse() method, which converts a JSON string into a JavaScript object.

- JSON.parse(window.localStorage.getItem('settings'));

Loop over keys of the localStorage object

- To iterate over name-value pairs stored in the localStorage, you use the Object.keys() method with for...of loop:

```
let keys = Object.keys(localStorage);
for(let key of keys) {
  console.log(`${key}: ${localStorage.getItem(key)}`);
}
```

Get Name of Key in Local Storage

- The key() method comes in handy in situations where you need to loop through keys and allows you to pass a number or index to localStorage to retrieve the name of the key.

```
var a = localStorage.key(1);
console.log(a);
```

Remove Value

- To delete local storage sessions, use the removeItem() method.

- When passed a key name, the removeItem() method removes that key from the storage if it exists. If there is no item associated with the given key, this method will do nothing.

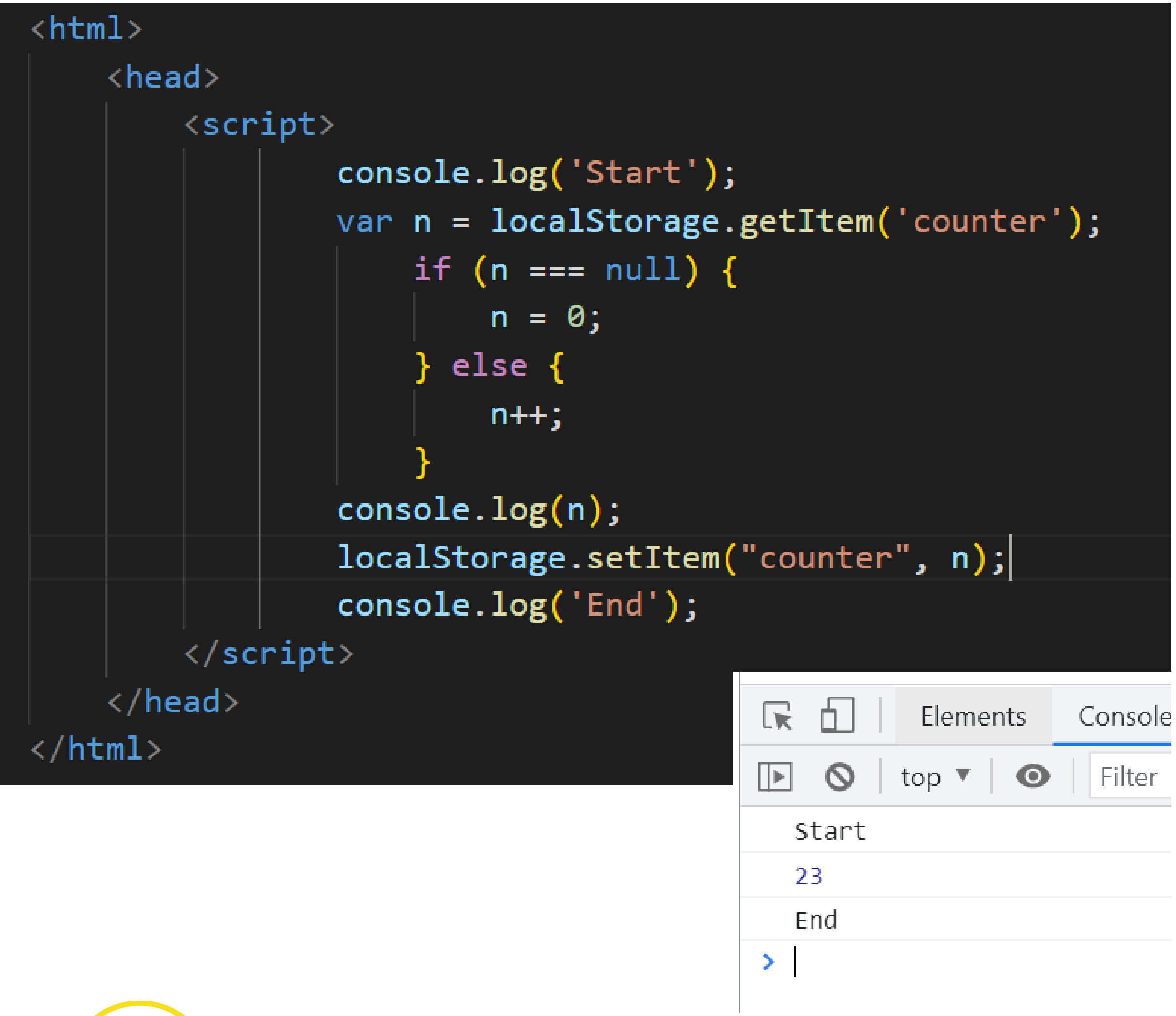
```
localStorage.removeItem('name')
```

How to delete all items in

- Use the clear() method to delete all items in localStorage.
- This method, when invoked, clears the entire storage of all records for that domain. It does not receive any parameters.

```
localStorage.clear();
```

Counter Example



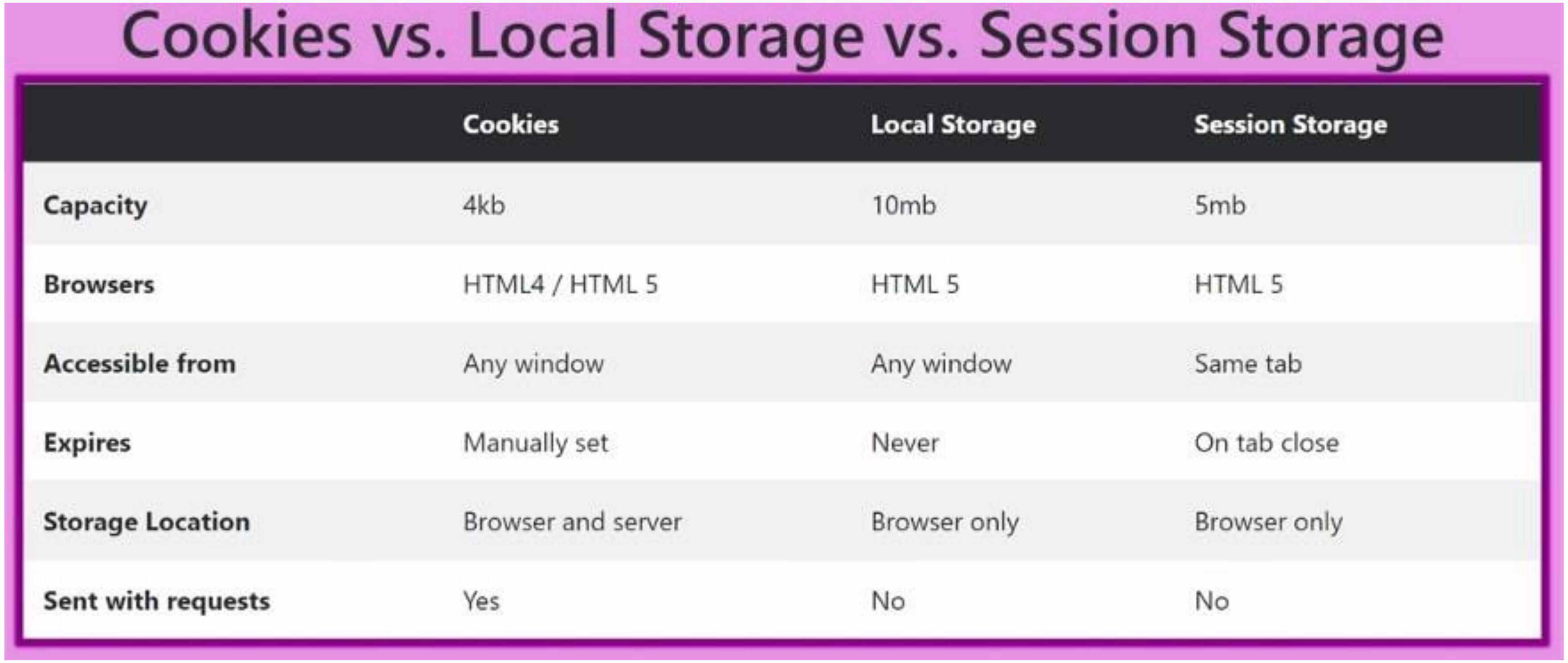
```
<html>
    <head>
        <script>
                console.log('Start');
                var n = localStorage.getItem('counter');
                    if (n === null) {
                        n = 0;
                    } else {
                        n++;
                    }
                console.log(n);
                localStorage.setItem("counter", n);
                console.log('End');
        </script>
    </head>
</html>
```

```
Start
23
End
>
```

```
<html>
    <head>
        <script>
                console.log('Start');
                var n = localStorage.getItem('counter');
                    if (n === null) {
                        n = 0;
                    } else {
                        n++;
                    }
                console.log(n);
                localStorage.setItem("counter", n);
                console.log('End');
        </script>
    </head>
</html>
```

sessionStorage

- The sessionStorage object stores data only for a session. It means that the data stored in the sessionStorage will be deleted when the browser is closed.
- A page session lasts as long as the web browser is open and survives over the page refresh.
- When you open a page in a new tab or window, the web browser creates a new session.
- When you close a tab or window, the web browser ends the session and clears data in the sessionStorage

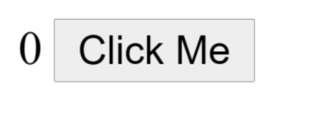


Cookies vs. Local Storage vs. Session Storage

	Cookies	Local Storage	Session Storage
Capacity	4kb	10mb	5mb
Browsers	HTML4 / HTML 5	HTML 5	HTML 5
Accessible from	Any window	Any window	Same tab
Expires	Manually set	Never	On tab close
Storage Location	Browser and server	Browser only	Browser only
Sent with requests	Yes	No	No

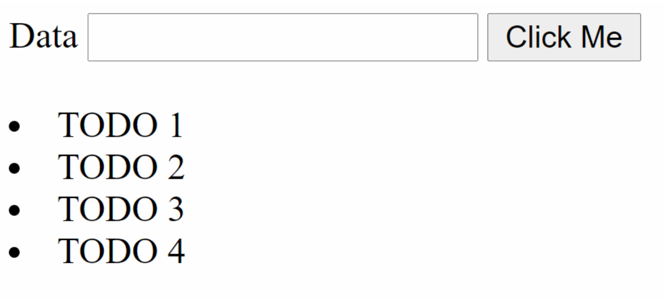
Task

- Counter Demo using Local Storage



0 Click Me

- TODO App Simple

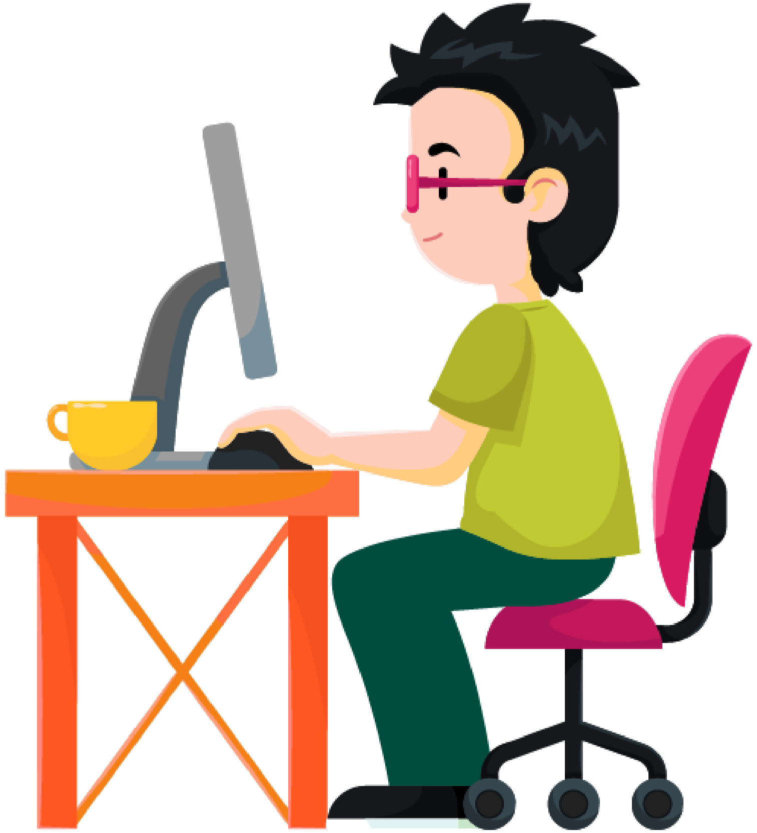


Data Click Me

- TODO 1
- TODO 2
- TODO 3
- TODO 4

- Todo App Using Local Storage
- CRUD Operation Using LocalStorage
  - Insert Form
  - Display Delete Edit

Get Exclusive Video Tutorials



www.aptutorials.com

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

Get More Details

www.akashsir.com

If You Liked It !

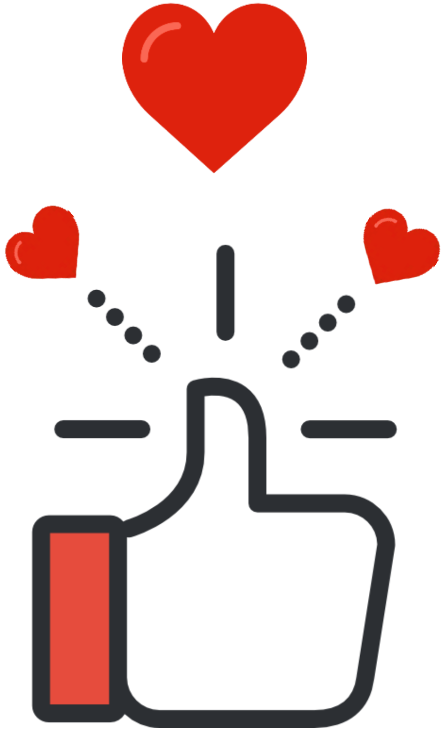
Rating Us Now

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.akashtechnolabs.com
www.akashpadhiyar.com
www.aptutorials.com

# Social Info

Akash.padhiyar

Akashpadhiyar

Akash_padhiyar

+91 99786-21654

#Akashpadhiyar
#aptutorials