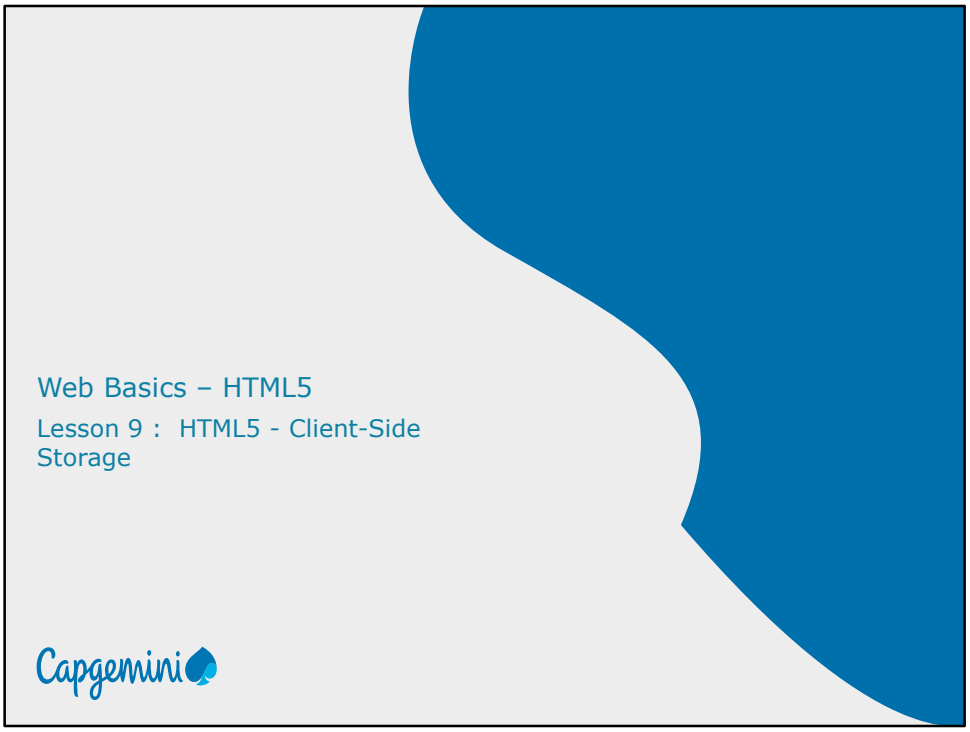
Instructor Notes:



Instructor Notes:

Lesson Objectives

➢**In this lesson you will learn about:**
- Introduction to HTML5 Client-Side Storage
- Types of Client-Side Storage

Instructor Notes:

Lesson Objectives

In this lesson you will learn about:

- Introduction to HTML5 Client-Side Storage
- Types of Client-Side Storage

Instructor Notes:

HTML5 Client-Side Storage – An Overview

- **Most talked about features in HTML 5**
- **Received a lot of criticism because of its lack of security, but it is nonetheless an interesting innovation**
- **Divided into 3 methodologies**
  - Session Storage
  - Local Storage
  - Database Storage

The HTML5 Web Storage (Client Storage):

The HTML5 (web) storage spec is a standardized way of providing larger amounts of client-side storage. Without HTML5, client-side storage for web applications is limited to the tiny storage provided by cookies i.e. 4KB per cookie, 20 cookies per domain. If cookies are used they provide both session and locally persistent storage at the same time, and are accessible by all browser windows and tabs. Domain cookies are sent with every request to that domain, which consumes bandwidth. The "mechanics" of processing cookies are also a bit cumbersome. In contrast, HTML5 storage provides a much larger initial local storage (5MB per domain), unlimited session storage (limited only by system resources) and successfully partitions local and session storage so that only the data you want to persist is persisted in local storage and data you want to be transient stays transient.

With HTML5 local storage, a larger amount of data (initially, 5MB per application per browser) can be persistently cached client-side, which provides an alternative to server downloads. A web application can achieve better performance and provide a better user experience if it uses this local storage. For example, your web application can use local storage to cache data from RPC calls for faster startup times and for a more responsive user interface. Other uses include saving the application state locally for a faster restore when the user re-enters the application, and saving the user's work if there is a network outage, and so forth

Instructor Notes:

Session Storage

- **Isn't much different from that what cookies offer, but has some additional benefits**
  - Session storage allows much more space, usually in megabytes
  - Depending on the browser implementation, the exact space can vary
  - Session data isn't sent automatically
  - Each tab/window maintains its own session information, as far as the site is concerned
- **Session Storage should be used to store short lived data related to a single browser window**
- **Data doesn't persist after the window is closed**
- **Methods for storing & retrieving data**
  - setItem(key,value): adds a key/value pair to the storage object
  - getItem(key): retrieves the value for a given key
  - clear(): removes all key/value pairs for the storage object
  - removeItem(key): removes a key/value pair from the storage object

Instructor Notes:

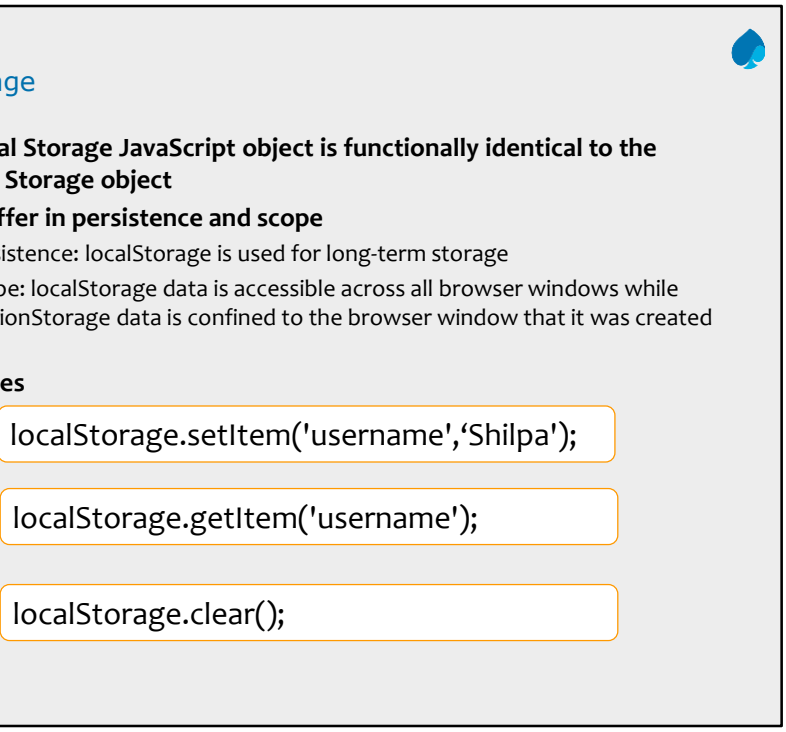
Session Storage - Example

- **Example**

```
sessionStorage.setItem('username',‘shilpa');
```

```
sessionStorage.getItem('username');
```

Instructor Notes:



Local Storage:
One LocalStorage per web application, with a max size of 5MB, is available for a given browser and is shared by all windows and tabs of that browser. For example, suppose you have MyWebApp running in a Chrome browser on the client. If you run MyWebApp in multiple tabs and windows, they all share the same Local Storage data ,subject to a max limit of 5MB. If you were to then open that same application in another browser, say FireFox, then the new browser would get its own LocalStorage to share with all its own tabs and windows.
HTML5 local storage saves data in string form as key-value pairs. If the data you wish to save is not string data, you are responsible for conversion to and from string when using LocalStorage.
HTML5 local storage saves data unencrypted in string form in the regular browser cache. It is not secure storage. It should not be used for sensitive data, such as social security numbers, credit card numbers, logon credentials, and so forth.

Instructor Notes:

Database Storage

- **When dealing with a larger amount of content, it would be nice to be able to store it in a structured manner and be able to access it randomly**
- **With HTML 5, you get database storage, which allows you to save structured data in the client's machine using a real SQL database**
- **Limitations**
  - Safari is the only browser to have implemented this feature with SQLite Database
  - No specifications on available SQL commands
  - The SQLite database also lacks the Foreign Key Constraint

Instructor Notes:

Database Storage

- **Current local database implementation is not quite fit for use, because of a few basic deficiencies such as**
  - Data is not encrypted
  - Accessible by anyone
  - There is no way to directly sync the local database with the one on the server
  - Limited memory space
- **Still, local database is going to be one of the key features in the future of browsers**

HTML5 Storage support for Session and Local storage	
Browser	Version
IE	8.0+
FIREFOX	3.5+
SAFARI	4.0+
CROME	4.0+
OPERA	10.5+
IPHONE	2.0+
ANDROID	2.0+

HTML5 Storage support for SQL DB	
Browser	Version
IE	
FIREFOX	
SAFARI	4.0+
CROME	4.0+
OPERA	10.5+
IPHONE	3.0+
ANDROID	2.0+

Instructor Notes:

Lab

- **Demonstration on HTML5 Client-Side Storage**

Instructor Notes:

Summary

In this module, you have learnt:

- The HTML5 (web) storage spec is a standardized way of providing larger amounts of client-side storage
- Without HTML5, client-side storage for web applications is limited to the tiny storage provided by cookies
- A web application can achieve better performance and provide a better user experience if it uses this local storage

Instructor Notes:

Summary

In this module, you have learnt:

- HTML5 Client-Side storage is divided into three categories
- Session Storage - Its similar to cookies but varies in size, accessible only withing the window or a tab that created it
- Local Storage - It can store 5MB per app per browser & deleted by user or by the app
- Database Storage - It provides good performance generally, being an asynchronous API.
- How to implement Client Side Storage in HTML5

Instructor Notes:

Review Question

- **Fill in the blank**
  - With HTML5 local storage, _____ amount of data per application per browser can be persistently cached client-side
  - _____ storage allows much more space, usually in megabytes
  - _____ method removes a key/value pair from the storage object
  - Safari is the only browser to have implemented this feature with _____ Database