

## Web Basics – HTML5

### Lesson 9 : HTML5 - Client-Side Storage



## Lesson Objectives

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



## **In this lesson you will learn about:**

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





## 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



# 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



# Session Storage - Example

## ➤ Example

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

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



# Local Storage

- **The local Storage JavaScript object is functionally identical to the session Storage object**
- **Only differ in persistence and scope**
  - Persistence: localStorage is used for long-term storage
  - Scope: localStorage data is accessible across all browser windows while sessionStorage data is confined to the browser window that it was created in
- **Examples**

```
localStorage.setItem('username','Shilpa');
```

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

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



# 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





# 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+

# Lab



## ➤ Demonstration on HTML5 Client-Side Storage





## **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





## **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



# 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

