UNIT 2: Create HTML5 document using advanced techniques with JavaScript and CSS3

Most Essential Learning Competencies: *At the end of the course, you must be able to:*

1. Discuss the fundamentals of E-Commerce
2. Explain and discuss local web storage
3. Add HTML5 local storage to shopping cart

Reading Activity

With the advent of HTML5, many sites were able to replace JavaScript plugin and codes with simple more efficient HTML codes such as audio, video, geolocation, etc. HTML5 tags made the job of developers much easier while enhancing page load time and site performance. In particular, HTML5 web storage was a game changer as they allow users' browsers to store user data without using a server. So the creation of web storage, allowed front-end developers to accomplish more on their website without knowing or using server-side coding or database.

Online e-commerce websites predominantly use server-side languages such as PHP to store users' data and pass them from one page to another. Using JavaScript back-end frameworks such as Node.js, we can achieve the same goal. However, in this tutorial, we'll show you step by step how to build a shopping cart with HTML5 and some minor JavaScript code. Other uses of the techniques in this tutorial would be to store user preferences, the user's favorite content, wish lists, and user settings like name and password on websites and native mobile apps without using a database.

Many high-traffic websites rely on complex techniques such as server clustering, DNS load balancers, client-side and server-side caching, distributed databases, and microservices to optimize performance and availability. Indeed, the major challenge for dynamic websites is to fetch data from a database and use a server-side language such as PHP to process them. However, remote database storage should be used only for essential website content, such as articles and user credentials. Features such as user preferences can be stored in the user's browser, similar to cookies. Likewise, when you build a native mobile app, you can use HTML5 web storage in conjunction with a local database to increase the speed of your app. Thus, as front-end developers, we need to explore ways in which we can exploit the power of HTML5 web storage in our applications in the early stages of development.

E-Commerce or Electronics Commerce is a methodology of modern business, which addresses the need of business organizations, vendors and customers to reduce cost and improve the quality of goods and services while increasing the speed of delivery. E-commerce refers to the paperless exchange of business information using the following ways:

- Electronic Data Exchange (EDI)
- Electronic Mail (e-mail)

- Electronic Bulletin Boards
- Electronic Fund Transfer (EFT)
- Other Network-based technologies

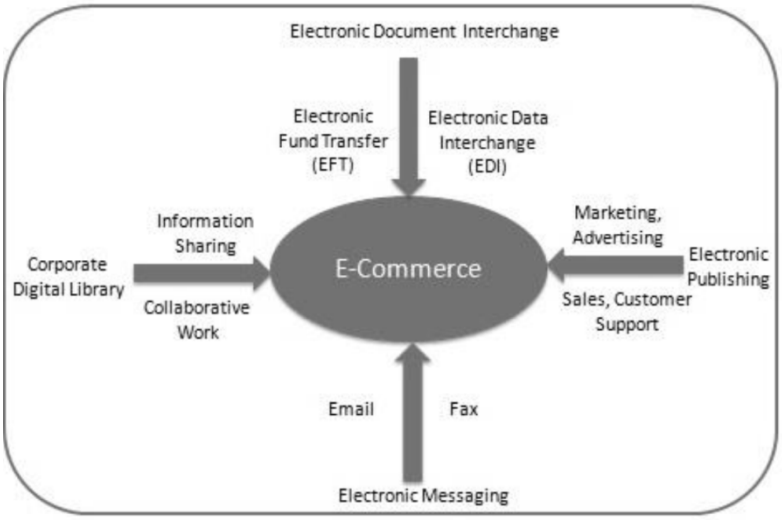


Figure 2-3.1: Electronic Document Interchange

E-Commerce provides the following features:

1. **Non-Cash Payment**: E-Commerce enables the use of credit cards, debit cards, smart cards, electronic fund transfer via bank's website, and other modes of electronics payment.
2. **24x7 Service availability**: E-commerce automates the business of enterprises and the way they provide services to their customers. It is available anytime, anywhere.
3. **Advertising/Marketing**: E-commerce increases the reach of advertising of products and services of businesses. It helps in better marketing management of products/services.
4. **Improved Sales**: Using e-commerce, orders for the products can be generated anytime, anywhere without any human intervention. It gives a big boost to existing sales volumes.
5. **Support**: E-commerce provides various ways to provide pre-sales and post-sales assistance to provide better services to customers.
6. **Inventory Management**: E-commerce automates inventory management. Reports get generated instantly when required. Product inventory management becomes very efficient and easy to maintain.
7. **Communication improvement**: E-commerce provides ways for faster, efficient, reliable communication with customers and partners.

Table 2-3.1: Traditional Versus E-Commerce

Traditional Commerce	E-Commerce
Heavy dependency on information exchange from person to person.	Information sharing is made easy via electronic communication channels making a little dependency on person to person information exchange.
Communication/transactions are done in synchronous way. Manual intervention is required for each communication or transaction.	Communication or transactions can be done in asynchronous way. The whole process is completely automated.
It is difficult to establish and maintain standard practices in traditional commerce.	A uniform strategy can be easily established and maintained in e-commerce.
Communications of business depends upon individual skills.	In e-commerce, there is no human intervention.
Unavailability of a uniform platform, as traditional commerce depends heavily on personal communication.	E-commerce websites provide the user a platform where all the information is available at one place.

The advantages of e-commerce can be broadly classified into three major categories:

1. Advantages to Organizations
2. Advantages to Consumers
3. Advantages to Society

Advantages to organization

Using e-commerce, organizations can expand their market to national and international markets with minimum capital investment. An organization can easily locate more customers, best suppliers, and suitable business partners across the globe.

- E-commerce helps organizations to reduce the cost to create process, distribute, retrieve and manage the paper based information by digitizing the information.
- E-commerce improves the brand image of the company.
- E-commerce helps organizations to provide better customer service.
- E-commerce helps to simplify the business processes and makes them faster and efficient.
- E-commerce reduces the paper work.
- E-commerce increases the productivity of organizations. It supports “pull” type supply management. In “pull” type supply management, a business process starts when a request comes from a customer and it uses just-in-time manufacturing way.

Advantages to Customers

- It provides 24x7 support. Customers can enquire about a product or service and place orders anytime, anywhere from any location.
- E-commerce application provides users with more options and quicker delivery of products.
- E-commerce application provides users with more options to compare and select the cheaper and better options.
- A customer can put review comments about a product and can see what others are buying, or see the review comments of other customers before making a final purchase.
- E-commerce provides options of virtual auctions.
- It provides readily available information. A customer can see the relevant detailed information within seconds, rather than waiting for days or weeks.
- E-Commerce increases the competition among organizations and as a result, organizations provides substantial discounts to customers.

Advantages to Society

- Customers need not travel to shop a product, thus less traffic on road and low air pollution.
- E-commerce helps in reducing the cost of products, so less affluent people can also afford the products.
- E-commerce has enabled rural areas to access services and products, which are otherwise not available to them.
- E-commerce helps the government to deliver public services such as healthcare, education, social services at a reduced cost and in an improved manner.

The disadvantages of e-commerce can be broadly classified into two major categories:

- Technical disadvantages
- Non-technical disadvantages

Technical Disadvantages

- There can be lack of system security, reliability or standards owing to poor implementation of e-commerce.
- The software development industry is still evolving and keeps changing rapidly.
- In many countries, network bandwidth might cause an issue.
- Special types of web servers or other software might be required by the vendor, setting the e-commerce environment apart from network servers.
- Sometimes, it becomes difficult to integrate an e-commerce software or website with existing applications or databases.
- There could be software/hardware compatibility issues, as some e-commerce software may be incompatible with some operating system or any other component.

Non-Technical Disadvantages

- Initial cost: The cost of creating/building an e-commerce application in-house may be very high. There could be delays in launching an e-Commerce application due to mistakes, and lack of experience.
- User resistance: Users may not trust the site being an unknown faceless seller. Such mistrust makes it difficult to convince traditional users to switch from physical stores to online/virtual stores.
- Security/ Privacy: It is difficult to ensure the security or privacy on online transactions.
- Lack of touch or feel of products during online shopping is a drawback.
- E-commerce applications are still evolving and changing rapidly.
- Internet access is still not cheaper and is inconvenient to use for many potential customers, for example, those living in remote villages.

E-commerce business models can generally be categorized into the following categories.

- Business – to – Business (B2B)
- Business – to – Consumer (B2C)
- Consumer – to – Consumer (C2C)
- Consumer – to – Business (C2B)
- Business – to – Government (B2G)
- Government – to – Business (G2B)
- Government – to – Citizen (G2C)

Business-to-Business

A website following the B2B business model sells its products to an intermediate buyer who then sells the product to the final customer. As an example, a wholesaler places an order from a company's website and after receiving the consignment, sells the end-product to the final customer who comes to buy the product at one of its retail outlets.

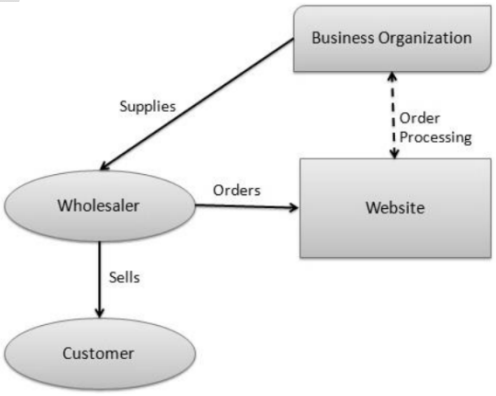


Figure 2-3.2: B2B

Business-to-Consumer

A website following the B2C business model sells its products directly to a customer. A customer can view the products shown on the website. The customer can choose a product and order the same. The website will then send a notification to the business organization via email and the organization will dispatch the product/goods to the customer.

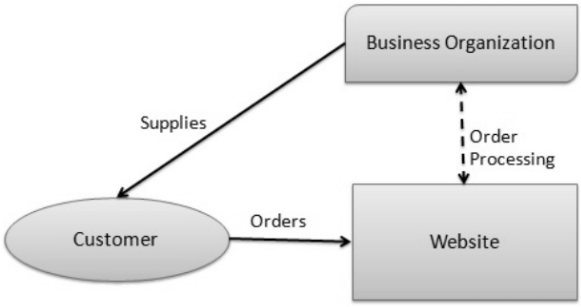


Figure 2-3.3: B2C

Consumer-to-Consumer

A website following the C2C business model helps consumers to sell their assets like residential property, cars, motorcycles, etc., or rent a room by publishing their information on the website. Website may or may not charge the consumer for its services. Another consumer may opt to buy the product of the first customer by viewing the post/advertisement on the website.

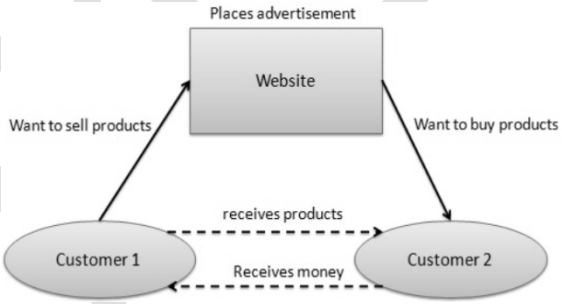


Figure 2-3.4: C2C

What is Web Storage?

With web storage, web applications can store data locally within the user's browser. Before HTML5, application data had to be stored in cookies, included in every server request. Web storage is more secure, and large amounts of data can be stored locally, without affecting website performance. Unlike cookies, the storage limit is far larger (at least 5MB) and information is never transferred to the server. Web storage is per origin (per domain and protocol). All pages, from one origin, can store and access the same data.

HTML5 web storage allows web applications to store values locally in the browser that can survive the browser session, just like cookies. Unlike cookies that need to be sent with every HTTP request, web

storage data is never transferred to the server; thus, web storage outperforms cookies in web performance. Furthermore, cookies allow you to store only 4 KB of data per domain, whereas web storage allows at least 5 MB per domain.

Browser Support

The numbers in the table specify the first browser version that fully supports Web Storage.

API					
Web Storage	4.0	8.0	3.5	4.0	11.5

HTML Web Storage Objects

HTML web storage provides two objects for storing data on the client:

- **window.localStorage** - stores data with no expiration date
- **window.sessionStorage** - stores data for one session (data is lost when the browser tab is closed)

The localStorage Object

The localStorage object stores the data with no expiration date. The data will not be deleted when the browser is closed, and will be available the next day, week, or year.

Example

```
// Store
localStorage.setItem("lastname", "Smith");

// Retrieve
document.getElementById("result").innerHTML = localStorage.getItem("lastname");
```

Example explained:

- Create a localStorage name/value pair with name="lastname" and value="Smith"
- Retrieve the value of "lastname" and insert it into the element with id="result"

The sessionStorage Object

The sessionStorage object is equal to the localStorage object, **except** that it stores the data for only one session. The data is deleted when the user closes the specific browser tab.

The following example counts the number of times a user has clicked a button, in the current session:

Example

```
if (sessionStorage.clickcount) {
  sessionStorage.clickcount = Number(sessionStorage.clickcount) + 1;
} else {
  sessionStorage.clickcount = 1;
}
document.getElementById("result").innerHTML = "You have clicked the button " +
sessionStorage.clickcount + " time(s) in this session.";
```

Self-Check

Quiz 2-3.1

Instructions: Write your answer on the Answer Sheet (AS) provided in this module (1-point each).

Enumerations:

1-5. E-commerce refers to the paperless exchange of business information using the following ways:

6-12. E-Commerce provides the following features:

13-15. Advantages of e-commerce can be broadly classified into three major categories:

Laboratory Activity

Activity 2-3.1

BUILD A BASIC SHOPPING CART

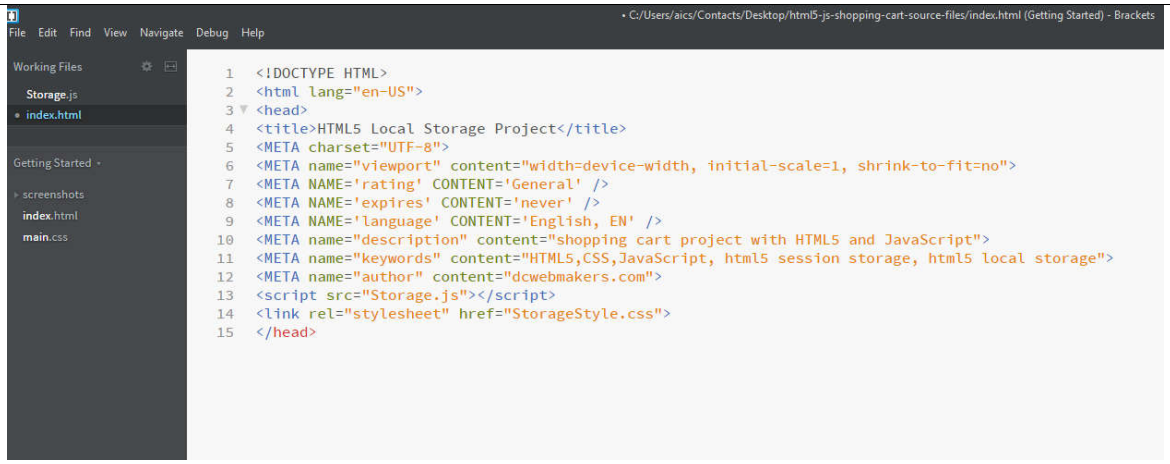
To build our shopping cart, we first create an HTML page with a simple cart to show items, and a simple form to add or edit the basket. Then, we add HTML web storage to it, followed by JavaScript coding. Although we are using HTML5 local storage tags, all steps are identical to those of HTML5 session storage and can be applied to HTML5 session storage tags. Lastly, we'll go over some jQuery code, as an alternative to JavaScript code, for those interested in using jQuery.

Adding HTML5 page

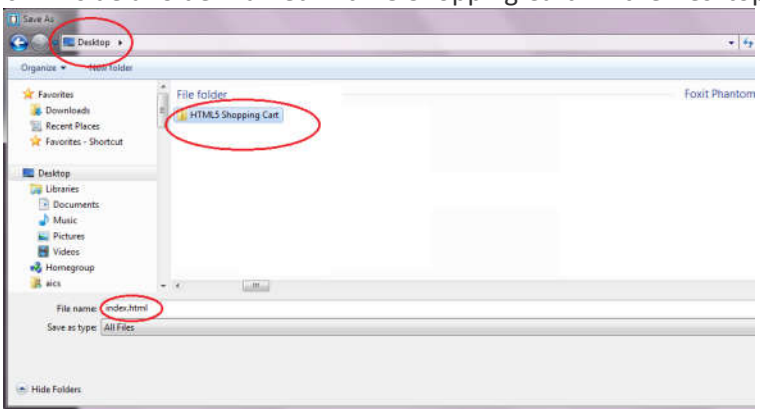
1. Open **Brackets** (code for the web) text editor or any text editor of your choice.

2. Copy-paste the html codes with tags for external JavaScript and CSS referenced in the head.

```
<!DOCTYPE HTML>
<html lang="en-US">
<head>
<title>HTML5 Local Storage Project</title>
<META charset="UTF-8">
<META name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<META NAME='rating' CONTENT='General' />
<META NAME='expires' CONTENT='never' />
<META NAME='language' CONTENT='English, EN' />
<META name="description" content="shopping cart project with HTML5 and JavaScript">
<META name="keywords" content="HTML5,CSS,JavaScript, html5 session storage, html5 local storage">
<META name="author" content="dcwebmakers.com">
<script src="Storage.js"></script>
<link rel="stylesheet" href="StorageStyle.css">
</head>
```

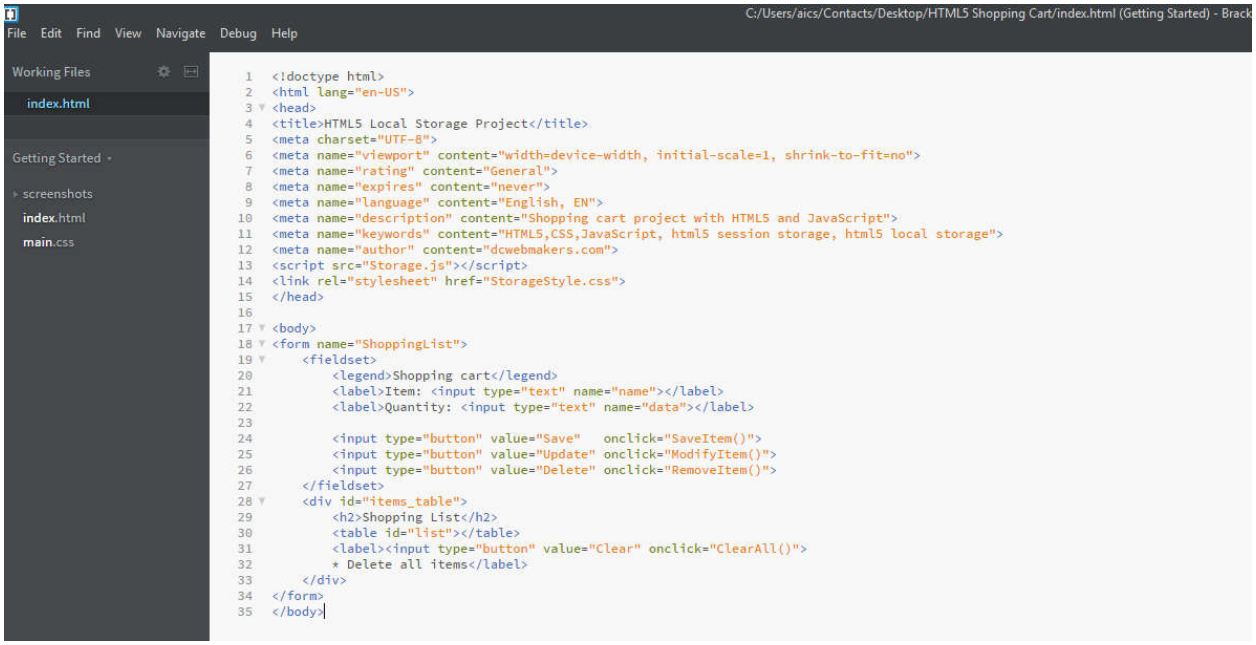


3. Save file as **index.html** inside a folder named “html5 Shopping Cart” in the Desktop directory.



4. Add the codes below inside the body tags

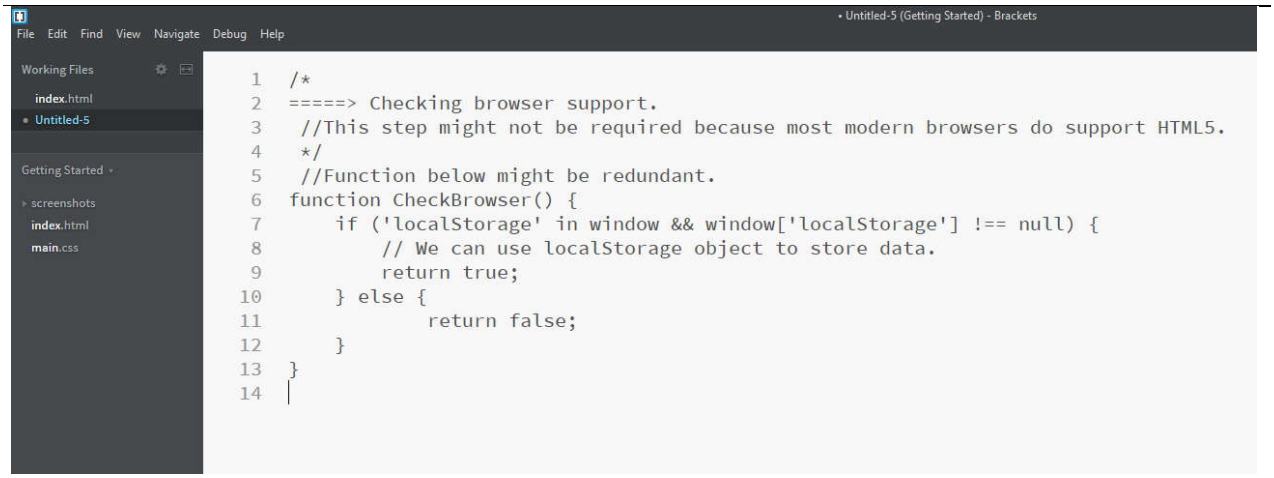
```
<body>
<form name="ShoppingList">
  <fieldset>
    <legend>Shopping cart</legend>
    <label>Item: <input type="text" name="name"></label>
    <label>Quantity: <input type="text" name="data"></label>

    <input type="button" value="Save"   onclick="SaveItem()">
    <input type="button" value="Update" onclick="ModifyItem()">
    <input type="button" value="Delete" onclick="RemoveItem()">
  </fieldset>
  <div id="items_table">
    <h2>Shopping List</h2>
    <table id="list"></table>
    <label><input type="button" value="Clear" onclick="ClearAll()">
    * Delete all items</label>
  </div>
</form>
</body>
```



Adding JavaScript functions

5. Create a CheckBrowser() function support in JavaScript.
6. Click File -> New in Brackets and copy-paste the code below

```
/*
 =====> Checking the browser support
 //this step may not be required as most of modern browsers do support HTML5
 */
 //below function may be redundant
function CheckBrowser() {
        if ('localStorage' in window && window['localStorage'] !== null) {
                // we can use localStorage object to store data
                return true;
        } else {
                        return false;
        }
}
```

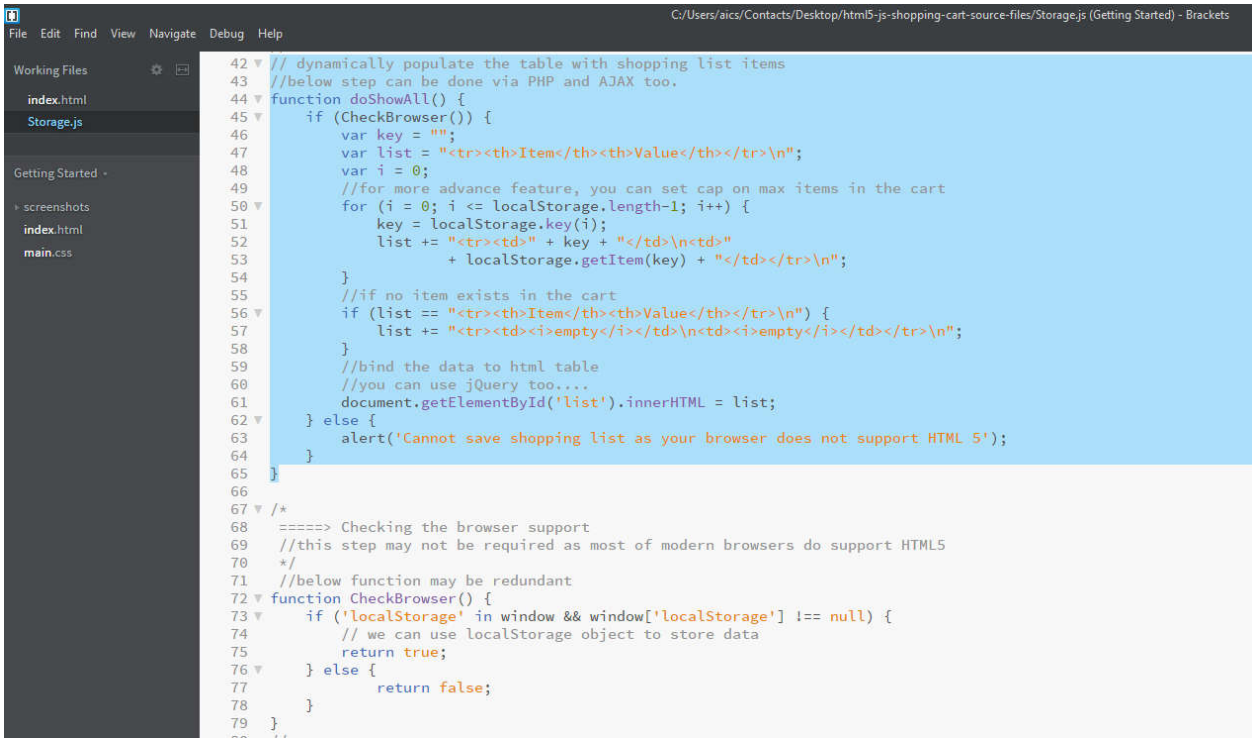


7. Save file as **Storage.js** inside the HTML5 Shopping Cart folder in the Desktop.

In the CheckBrowser() function, we would like to check whether the browser supports HTML5 storage. Note that this step might not be required because most modern web browsers support it.

8. Create a doShowAll() functions
9. Copy-paste the codes above the CheckBrowser() function.

```
// Dynamically populate the table with shopping list items.
//Step below can be done via PHP and AJAX, too.
function doShowAll() {
   if (CheckBrowser()) {
      var key = "";
      var list = "<tr><th>Item</th><th>Value</th></tr>\n";
      var i = 0;
      //For a more advanced feature, you can set a cap on max items in the cart.
      for (i = 0; i <= localStorage.length-1; i++) {
         key = localStorage.key(i);
         list += "<tr><td>" + key + "</td>\n<td>"
               + localStorage.getItem(key) + "</td></tr>\n";
      }
      //If no item exists in the cart.
      if (list == "<tr><th>Item</th><th>Value</th></tr>\n") {
         list += "<tr><td><i>empty</i></td>\n<td><i>empty</i></td></tr>\n";
      }
      //Bind the data to HTML table.
      //You can use jQuery, too.
      document.getElementById('list').innerHTML = list;
   } else {
      alert('Cannot save shopping list as your browser does not support HTML 5');
   }
}
```

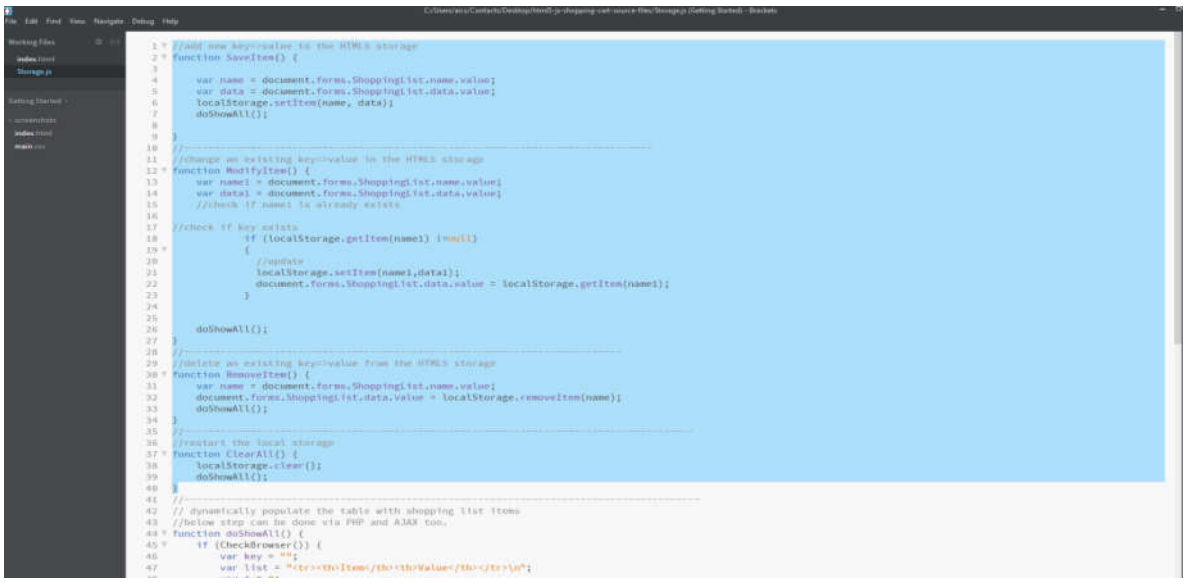


Inside the **doShowAll(),** if the **CheckBrowser()** function evaluates first for browser support, then it will dynamically create the table for the shopping list during page load. You can iterate the keys (property names) of the key-value pairs stored in local storage inside a JavaScript loop, as shown below. Based on the storage value, this method populates the table dynamically to show the key-value pair stored in local storage.

10. Copy-paste the code (on top of the existing codes)

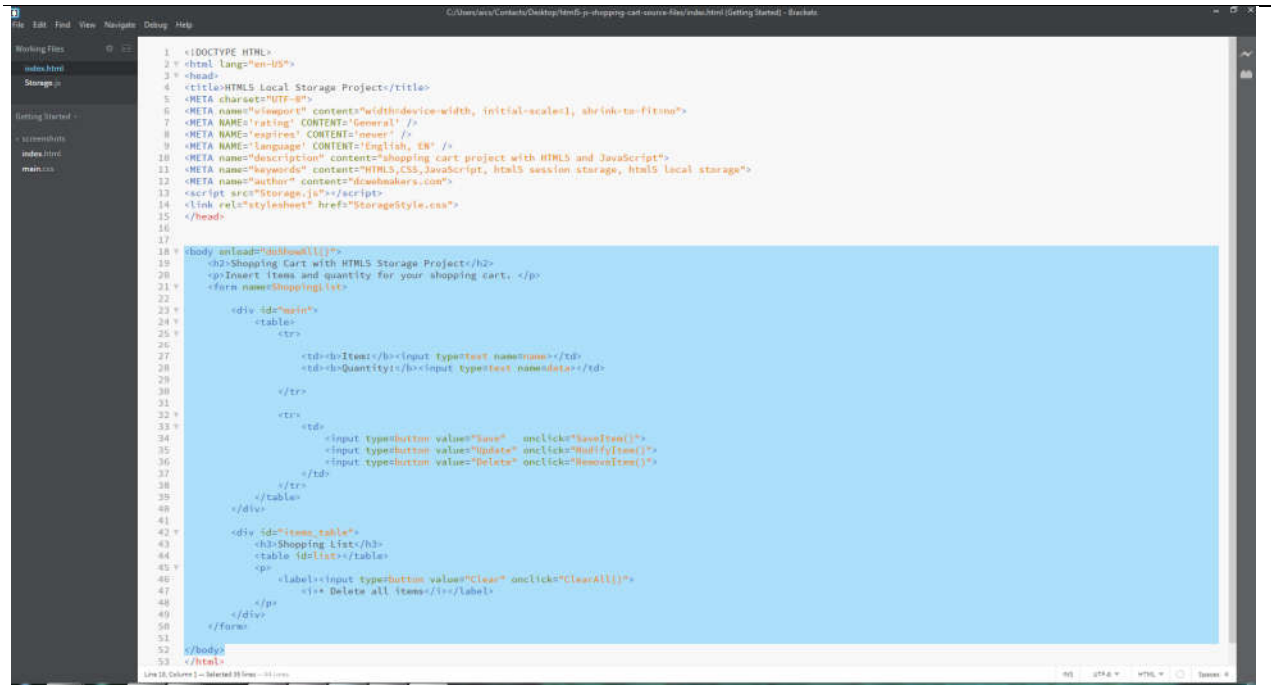
```
//add new key=>value to the HTML5 storage
function SaveItem() {

        var name = document.forms.ShoppingList.name.value;
        var data = document.forms.ShoppingList.data.value;
        localStorage.setItem(name, data);
        doShowAll();

}
//------------------------------------------------------------------------
//change an existing key=>value in the HTML5 storage
function ModifyItem() {
        var name1 = document.forms.ShoppingList.name.value;
        var data1 = document.forms.ShoppingList.data.value;
        //check if name1 is already exists
```

```
//check if key exists
                        if (localStorage.getItem(name1) !=null)
                        {
                          //update
                          localStorage.setItem(name1,data1);
                          document.forms.ShoppingList.data.value =
localStorage.getItem(name1);
                        }


        doShowAll();
}
//-------------------------------------------------------------------------
//delete an existing key=>value from the HTML5 storage
function RemoveItem() {
        var name = document.forms.ShoppingList.name.value;
        document.forms.ShoppingList.data.value = localStorage.removeItem(name);
        doShowAll();
}
//-----------------------------------------------------------------------------------
//restart the local storage
function ClearAll() {
        localStorage.clear();
        doShowAll();
}
```

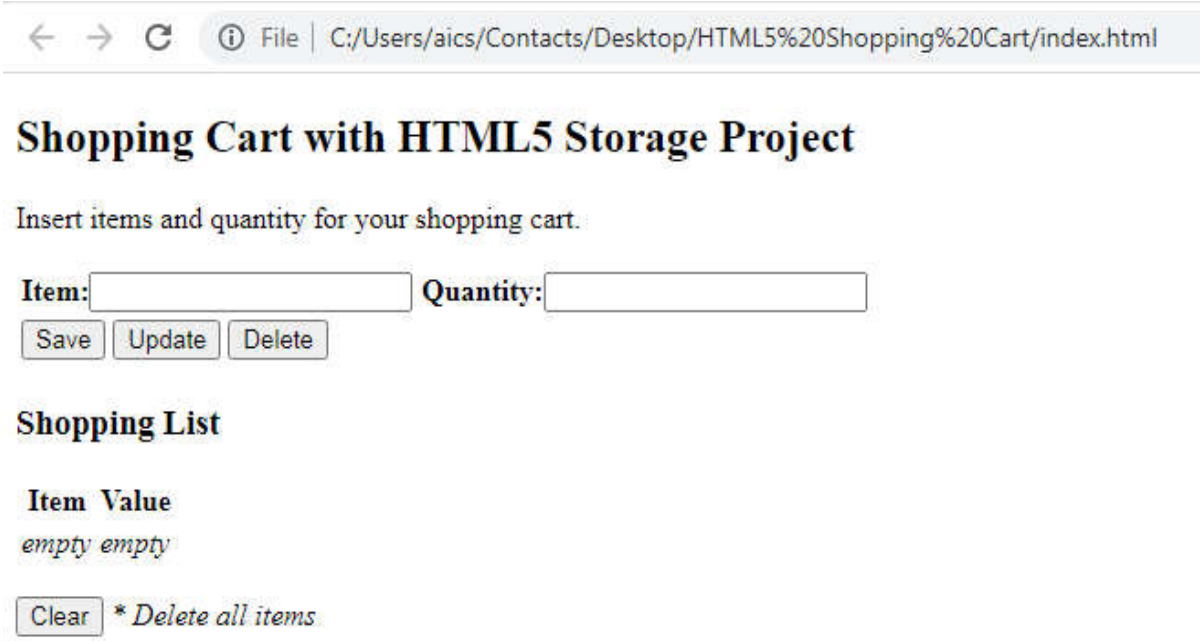


11. Click File ->Save
12. Switch to index.html
13. Update the existing body section with the new codes. Copy-paste the code (index.html)

```
<body onload="doShowAll()">
	<h2>Shopping Cart with HTML5 Storage Project</h2>
	<p>Insert items and quantity for your shopping cart. </p>
	<form name=ShoppingList>

		<div id="main">
			<table>
				<tr>

					<td><b>Item:</b><input type=text name=name></td>
					<td><b>Quantity:</b><input type=text
name=data></td>

				</tr>

				<tr>
					<td>
					  <input type=button value="Save"
onclick="SaveItem()">
					  <input type=button value="Update"
onclick="ModifyItem()">
					  <input type=button value="Delete"
onclick="RemoveItem()">
					</td>
				</tr>
			</table>
		</div>

		<div id="items_table">
			<h3>Shopping List</h3>
			<table id=list></table>
			<p>
				<label><input type=button value="Clear" onclick="ClearAll()">
					<i>* Delete all items</i></label>
			</p>
		</div>
	</form>

</body>
```



14. Run and test the shopping cart



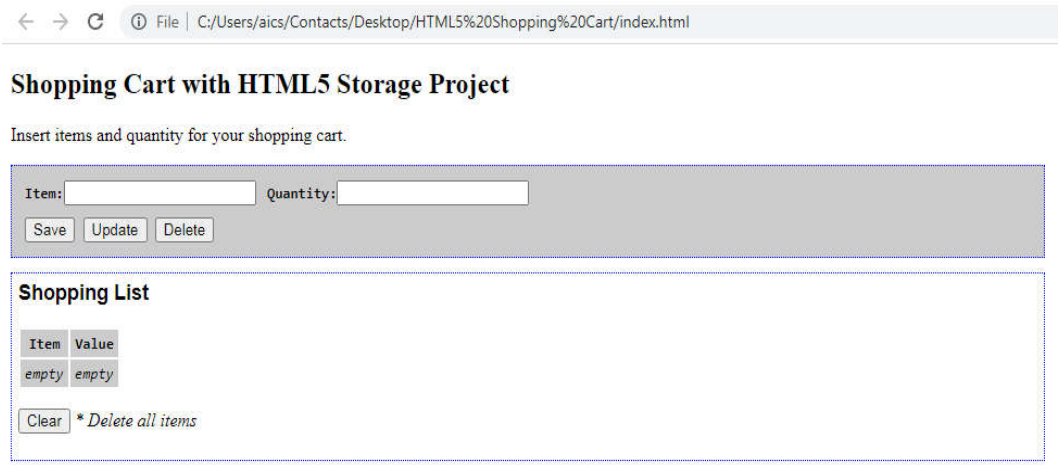
15. Input some data in the Item and Quantity fields. Try to click the Save, Update, Delete and Clear buttons to see what happens.

Adding CSS to the HTML page

16. Click File -> New
17. Copy-paste the code

```
td,th {
	font-family: monospace;
	padding: 4px;
	background-color: #ccc;
}

label {
	vertical-align: top;
}

#main {
	border: 1px dotted blue;
	padding: 6px;
	background-color: #ccc;
	margin-right: 50%;
}

#items_table {
	border: 1px dotted blue;
	padding: 6px;
	margin-top: 12px;
	margin-right: 50%;
}

#items_table h3 {
	font-size: 18px;
	margin-top: 0px;
	font-family: sans-serif;
}
```

18. Save the file as **"StorageStyle.css"** inside the HTML5 Shopping Cart folder in the Desktop directory.
19. Run **index.html**



- ***End of Activity –***

-

Read Additional Resources

HTML5 Cheat Sheets

1. Week2-3_Introduction-to-e-commerce.pdf

Watch Video Resources

1. Week2-3_What is Ecommerce and How Does it Work.mp4
2. Week2-3_ How A Shopping Cart Works.mp4
3. Week2-3_ How HTML5 Local Storage Works.mp4

Internet References

1. https://w3schools.com
2. https://www.smashingmagazine.com/2019/08/shopping-cart-html5-web-storage/
3. https://www.tutorialspoint.com/e_commerce/e_commerce_tutorial.pdf
4. http://www. youtube.com
5. https://www.w3schools.com/html/html5_webstorage.asp
6. https://www.smashingmagazine.com/2019/08/shopping-cart-html5-web-storage/