

**Week 9 Homework part 2**

1. What is the Web Storage API?

Web Storage API allows applications to store data within an individual user's browser.

2. What data storage was used before HTML5 was introduced? What was the data storage called and what was it included in?

The data storage that was used before HTML5 was cookies. Cookies were included in each server request.

3. What are 3 advantages of using the Web Storage API over cookies?

- Web Storage is more secure and data is never transferred to the server.
- Large amounts of data can be stored locally without affecting website performance and stored data has no expiration date by default. It can only be cleared by using JavaScript, clearing the Browser cache, or localStorage.
- The storage limit is the maximum compared to sessionStorage and cookies

4. What are the two objects called for storing data on the client that HTML5 Web Storage provides?

localStorage and sessionStorage

5. Can localStorage and sessionStorage be read server-side?

No, only client-side

6. Which data item can be read server-side? Why is it popular? What is it used mostly for? And what kind of data might it contain?

Cookies read server-side. They are popular because they allow sites to store user data directly into the user's web browser. It is usually used for server-side reading.

7. Why do cookies have to be well secured and protected?

Cookies can contain sensitive data so they have to be protected.

8. What is the `httpOnly` flag good for? And what does it help mitigate the negative impact of, for example?

Cookies can be protected and secured by setting the `httpOnly` flag to equal `true`. This blocks client-side access to that cookie. It helps mitigate the negative impact of the XSS attack, which is a Cross-Site Scripting attack.

9. What are `localStorage` and `sessionStorage` of the Web Storage API connected to? And what does that connection make possible for pages from the same origin to do?

Web Storage, local or session, is connected to a particular origin, which can be a domain or protocol. All pages from one origin can store and access the same data.

10. What does the Storage Interface of the Web Storage API permit?

It permits you to set, retrieve, and remove data for a specific domain and storage type.

11. The Storage Object, represented by the `localStorage` and `sessionStorage` objects, contains one property. What is that property called and what does it return? Use your project application to describe what this property does.

The Storage Objects (`localStorage` and `sessionStorage`) contains `storage.length`, the length property which returns an integer representing the number of data items stored in the Storage object.

12. Give me the five Storage methods of the Web Storage API and describe what each does. When applicable, use code from your project application as examples.

- `Storage.removeItem()` - when passed, a key name will remove that key from the storage. If not, it is associated with the given key then this method will do nothing.

Eg: `function clearStorage() {`

```
textField.value = "";

sessionStorage.removeItem("autosave", textField.value);}
```

- `Storage.setItem()` - when passed, key name and value will add that key to the storage/update that key's value if it already exists.

Eg: `try {`

```
localStorage.setItem("autosave", textField.value);}
```

- `Storage.clear()` - when invoked, will empty all keys out of storage.

Eg: `function emptyStorage() {`

```
textField.value = "";  
  
sessionStorage.clear();}
```

- `Storage.getItem()` - when passed a key name will return that key's value or null if the key is non-existent.

Eg: no example from my code

- `Storage.key()` - when passed a certain number represented by `n`, this method returns the name of the `n`th key in the storage.

Eg: no example from my code

13. When using cookies, why would you want to use the Secure flag?

Only the Secure attribute lets you forbid a cookie to be ever transmitted over simple HTTP. It can only be transmitted over HTTPS.

14. What is the difference between `localStorage` and `sessionStorage`?

The differences are that data in `localStorage` doesn't expire, data in `sessionStorage` is cleared when the page session ends. A page session lasts as long as the browser is open, and survives over page reloads and restores.

15. What are the similarities between localStorage and sessionStorage?

The data is never transferred to the server, their storage limits are larger than cookies, and have the same API.

16. Explain what try...catch is and what it does.

The try...catch statement marks a block of statements to try out code and specifies a response if an exception is thrown.

17. Why do we have to set a try...catch statement in our application? Refer to your/this homework README.md for the answer.

If there is sessionStorage support, then the program steps into the else statement. There we set and get our sessionStorage as well as check if we have exceeded our sessionStorage quota. Since we have to catch our exception, we need to use a try...catch statement.

18. Explain what JSON.stringify() does and provide an example using code from your project application.

JSON.stringify() method converts a JavaScript object or value to a JSON string. We take the sessionStorage object's autosave property's value and extract it from sessionStorage and transfer the value into a string so that we may insert it into a .txt file.

19. Describe what the value of the csv variable is and why it is important for our application.

CSV variable is new Blob(blobParts options). The BlobParts: an array of the Blob bufferSource or strings values. This is important for our application so that we can input a value to be transferred into a string. This then is put into a .txt file for users to download.

20. Please copy and paste the result of the console.log(csv) after you have clicked on the Save Text To File For Download button. Hint: you should try this out only after you have completed building your application.

Session.js:10 "SUP BOIBOI do you work? OKay!!" (what I wrote down I the textboxarea)