Experiment No. 8

Aim : To code and register a service worker, and complete the install and activation process for a new service worker for the E-commerce PWA.

Theory :

Service Worker

Service Workers play a crucial role in Progressive Web Applications (PWAs) by acting as a proxy between the web application and the network. This allows them to provide offline capabilities, background data synchronization, and push notifications, enhancing the app's performance, reliability, and user experience.

- **Network Proxy:** Service workers intercept all outgoing HTTP requests, allowing you to handle these requests. For example, you can serve content from a local cache if available, improving performance and providing a better user experience.
- **Offline Capabilities:** Service workers enable offline functionality by caching essential application resources such as HTML, CSS, JavaScript, and images. When a user is offline, the service worker can retrieve the requested content from the cache, ensuring a seamless experience even without an internet connection.
- **HTTPS Requirement:** For security reasons, service workers require HTTPS connections. This ensures secure communication between the service worker, your application, and the server.

What can't we do with Service Workers?

- You can't access the **Window**
  You can't access the window, therefore, You can't manipulate DOM elements. But, you can communicate to the window through post Message and manage processes that you want.

- You can't work it on **80 Port**
  Service Worker just can work on HTTPS protocol. But you can work on localhost during development.

What can we do with Service Workers?

- **Network Traffic Control:** Service Workers provide the ability to regulate all network traffic within a webpage, granting the capability to manipulate requests and responses. For instance, one can respond to a request for a CSS file with plain text or to an HTML file request with a PNG file. Moreover, it allows for customized responses according to the requested content.
- **Caching:** Leveraging the Service Worker and Cache API, developers can cache any request/response pair. This facilitates accessing offline content seamlessly, ensuring that users can still interact with the application even without an active internet connection.
- **Push Notifications:** Service Workers enable the management of push notifications, facilitating the display of informative messages to users. This feature enhances user engagement by keeping them informed about relevant updates or events, even when they are not actively using the application.
- **Background Processes:** With the Background Sync feature of Service Workers, developers can initiate processes even in situations where the internet connection is disrupted. This ensures that critical tasks or updates can be queued and executed once the connection is restored, providing a seamless user experience.

Service Worker Cycle

Steps for coding and registering a service worker for your E-commerce PWA completing the install and activation process:

Name: Roshni Talreja **Expt no: 08** **Roll no: 66**

1. Create the Service Worker File (sw.js):

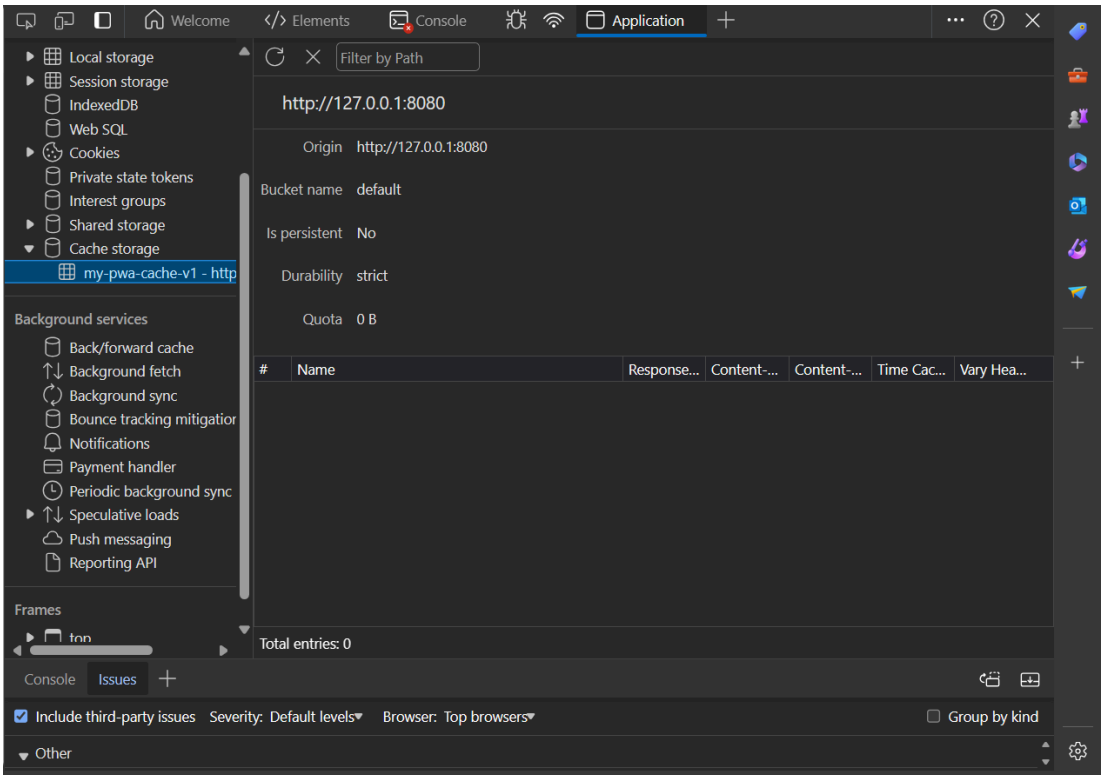
```
// sw.js
const cacheName = 'my-pwa-cache-v1';
const staticAssets = [
    '/',
    '/index.html',
    '/style.css',
    '/script.js'
];

self.addEventListener('install', event => {
    event.waitUntil(
        caches.open(cacheName)
            .then(cache => {
                return cache.addAll(staticAssets);
            })
    );
});

self.addEventListener('fetch', event => {
    event.respondWith(
        caches.match(event.request)
            .then(response => {
                return response || fetch(event.request);
            })
    );
});
```

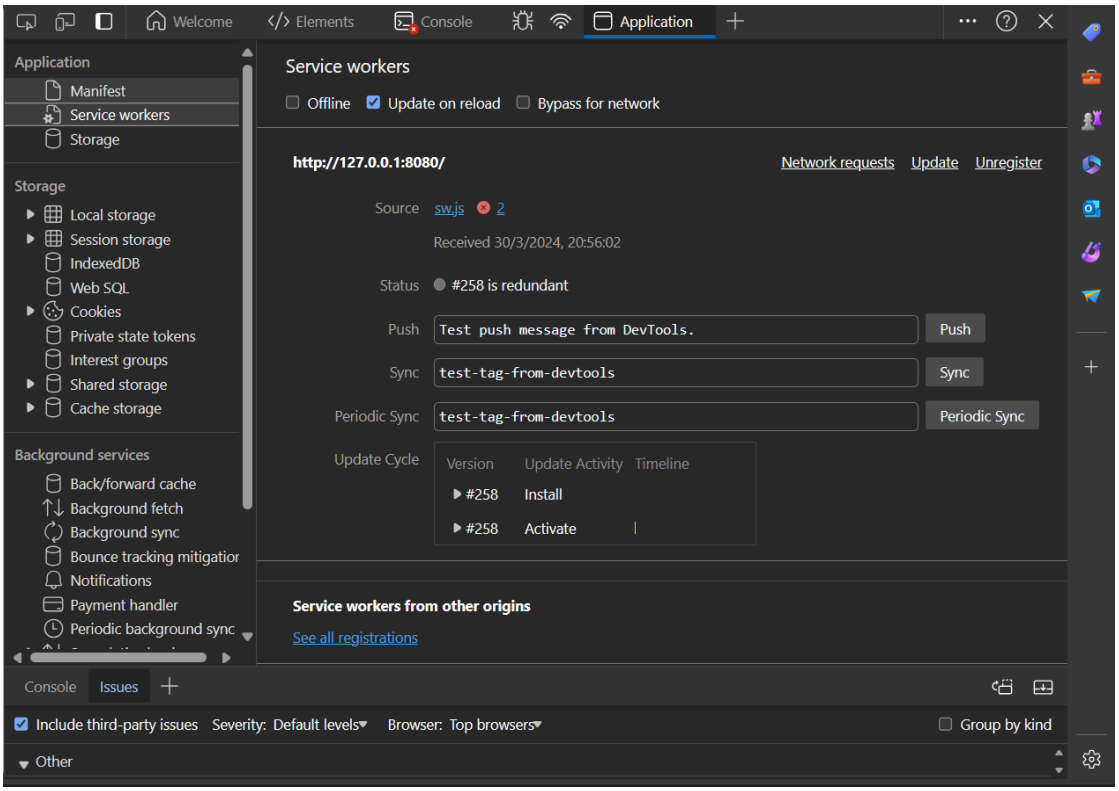
2. Register the Service Worker:

In your main JavaScript file (e.g., `script.js`), add the following code:

Output

Name: Roshni Talreja **Expt no: 08** **Roll no: 66**





Conclusion : I have understood and successfully registered a service worker, and completed the install and activation process for a new service worker for the E-commerce PWA.