1. Aim

To code and register a service worker, and complete the install and activation process for a new service worker in an E-commerce Progressive Web App (PWA).

2. Basic Description

A **Progressive Web App (PWA)** uses modern web capabilities to deliver an app-like experience to users. PWAs are reliable, fast, and engaging. One of the core technologies behind a PWA is the **Service Worker**.

A **Service Worker** is a JavaScript file that runs in the background, separate from the main browser thread. It acts like a network proxy that enables features like:

- **Offline support**
- **Caching assets**
- **Push notifications**
- **Background sync**

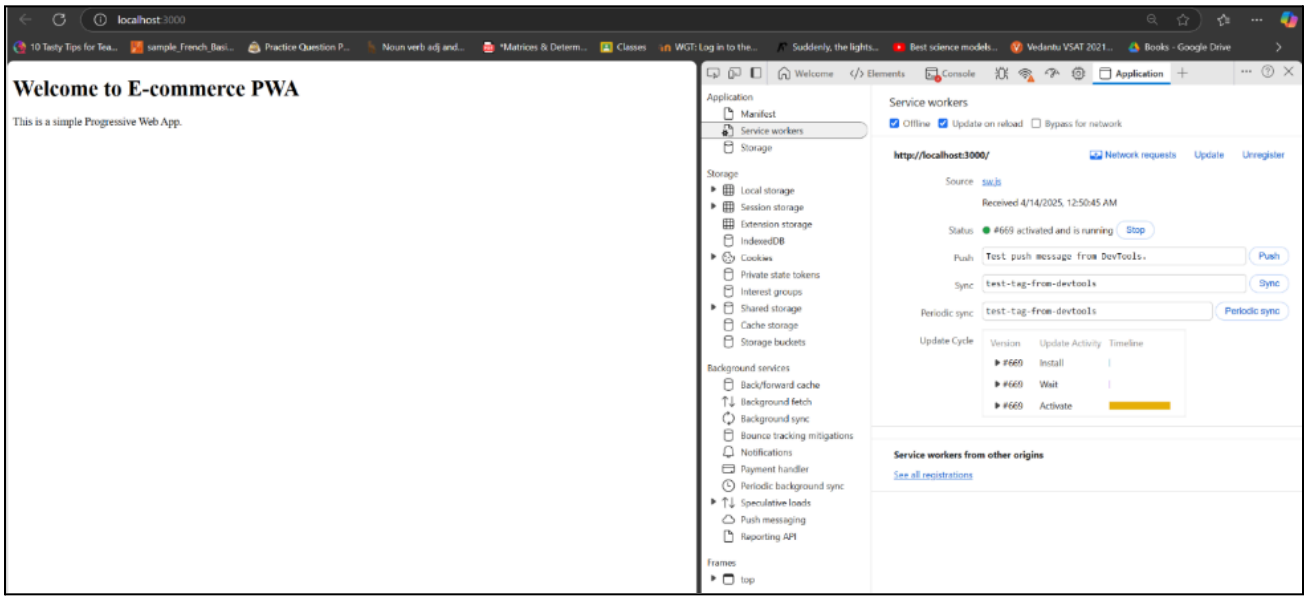
Lifecycle of a Service Worker:

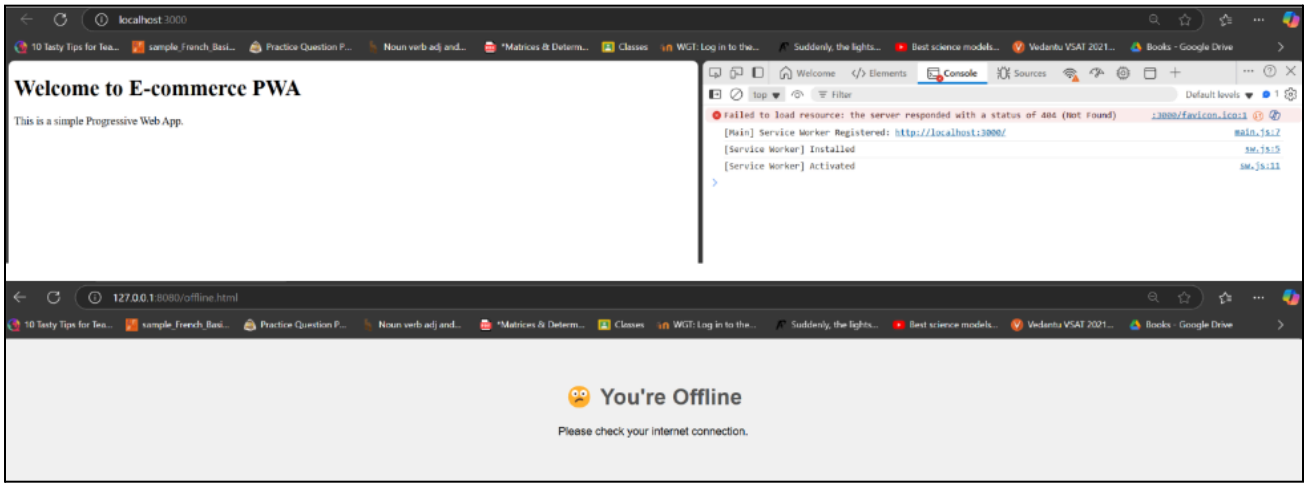
1. **Installation** – Triggered when the service worker is first registered.
2. **Activation** – Occurs after installation; prepares the worker to control pages.
3. **Fetch** – Intercepts network requests and serves them from the cache or network.

How it works in our E-Commerce PWA:

- When a user visits the website, the service worker is registered.
- It caches essential files like index.html, style.css, manifest.json, and product images.
- On subsequent visits, it loads content from the cache, ensuring faster performance and offline access.

3. Output:





4. Conclusion

By implementing a service worker in the E-Commerce PWA, we have added offline capabilities and enhanced loading speed, making the web application more reliable and user-friendly. This experiment demonstrates how modern web technologies improve user experience in real-world scenarios like online shopping platforms.