

PRACTICAL 8

Name : Swarnika Singh

Class:D15B

Roll no: 63

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 Worker is a script that works on browser background without user interaction independently. Also, It resembles a proxy that works on the user side. With this script, you can track network traffic of the page, manage push notifications and develop “offline first” web applications with Cache API.

Things to note about Service Worker:

- A service worker is a programmable network proxy that lets you control how network requests from your page are handled.
- Service workers only run over HTTPS. Because service workers can intercept network requests and modify responses, "man-in-the-middle" attacks could be very bad.
- The service worker becomes idle when not in use and restarts when it's next needed.
You cannot rely on a global state persisting between events. If there is information that you need to persist and reuse across restarts, you can use IndexedDB databases.

What can we do with Service Workers?

- You can dominate Network Traffic
You can manage all network traffic of the page and do any manipulations. For example, when the page requests a CSS file, you can send plain text as a response or when the page requests an HTML file, you can send a png file as a response. You can also send a true response too.
- You can Cache
You can cache any request/response pair with Service Worker and Cache API and you can access these offline content anytime.
- You can manage Push Notifications
You can manage push notifications with Service Worker and show any information message to the user.
- You can Continue
Although Internet connection is broken, you can start any process with Background Sync of Service Worker.

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.

Service Worker Cycle

A service worker goes through three steps in its life cycle:

- Registration
- Installation
- Activation

Registration

To install a service worker, you need to register it in your main JavaScript code. Registration tells the browser where your service worker is located, and to start installing it in the background. Let's look at an example:

main.js

```
if ('serviceWorker' in navigator) { navigator.serviceWorker.register('/service-worker.js')
.then(function(registration) {
console.log('Registration successful, scope is:', registration.scope);
})
.catch(function(error) {
console.log('Service worker registration failed, error:', error);
});
}
```

This code starts by checking for browser support by examining `navigator.serviceWorker`. The service worker is then registered with `navigator.serviceWorker.register`, which returns a promise that resolves when the service worker has been successfully registered. The scope of the service worker is then logged with `registration.scope`. If the service worker is already installed, `navigator.serviceWorker.register` returns the registration object of the currently active service worker.

The scope of the service worker determines which files the service worker controls, in other words, from which path the service worker will intercept requests. The default scope is the location of the service worker file, and extends to all directories below. So if `service-worker.js` is located in the root directory, the service worker will control requests from all files at this domain.

You can also set an arbitrary scope by passing in an additional parameter when registering. For example:

```
main.js
```

```
navigator.serviceWorker.register('/service-worker.js', { scope: '/app/'
});
```

In this case we are setting the scope of the service worker to `/app/`, which means the service worker will control requests from pages like `/app/`, `/app/lower/` and `/app/lower/lower`, but not from pages like `/app` or `/`, which are higher.

If you want the service worker to control higher pages e.g. `/app` (without the trailing slash) you can indeed change the scope option, but you'll also need to set the `Service-Worker-Allowed` HTTP Header in your server config for the request serving the service worker script.

```
main.js
```

```
navigator.serviceWorker.register('/app/service-worker.js', { scope: '/app'
});
```

Installation

Once the browser registers a service worker, installation can be attempted. This occurs if the service worker is considered to be new by the browser, either because the site currently doesn't have a registered service worker, or because there is a byte difference between the new service worker and the previously installed one.

A service worker installation triggers an `install` event in the installing service worker. We can include an `install` event listener in the service worker to perform some task when the service worker installs. For instance, during the `install`, service workers can precache parts of a web app so that it loads instantly the next time a user opens it (see caching the application shell). So, after that first load, you're going to benefit from instant repeat loads and your time to interactivity is going to be even better in those cases. An example of an installation event listener looks like this:

```
service-worker.js
```

```
// Listen for install event, set callback
self.addEventListener('install', function(event) {
// Perform some task
});
```

Activation

Once a service worker has successfully installed, it transitions into the activation stage. If there are any open pages controlled by the previous service worker, the new service worker enters a waiting state. The new service worker only activates when there are no longer any

pages loaded that are still using the old service worker. This ensures that only one version of the service worker is running at any given time.

When the new service worker activates, an activate event is triggered in the activating service worker. This event listener is a good place to clean up outdated caches (see the Offline Cookbook for an example).

service-worker.js

```
self.addEventListener('activate', function(event) {  
  // Perform some task  
});
```

Once activated, the service worker controls all pages that load within its scope, and starts listening for events from those pages. However, pages in your app that were loaded before the service worker activation will not be under service worker control. The new service worker will only take over when you close and reopen your app, or if the service worker calls `clients.claim()`. Until then, requests from this page will not be intercepted by the new service worker. This is intentional as a way to ensure consistency in your site.

CODE AND OUTPUT:

1) Index.html

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <!-- Google tag (gtag.js) -->  
<script async src="https://www.googletagmanager.com/gtag/js?id=G-CSQNB0SBN9"></script>  
<script>  
  window.dataLayer = window.dataLayer || [];  
  function gtag(){dataLayer.push(arguments);}  
  gtag('js', new Date());  
  
  gtag('config', 'G-CSQNB0SBN9');  
</script>  
  
<link rel="manifest"  
href="manifest.json">  
  
  <meta charset="UTF-8">  
  <meta http-equiv="X-UA-Compatible" content="IE=edge">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <meta name="theme-color" content="#4285f4">  
  <!-- CSS LINK -->  
  <link rel="stylesheet" href="style.css">  
  
  <!-- Fontawesome -->  
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.1/css/all.min.css"  
integrity="sha512-KfkfwYDsLkIlwQp6LFnl8zNdLGxu9YAA1QvwINks4PhcElQSVqcyVLLD9aMhXd13uQjoXtEKNosOWaZqXgel  
0g==" crossorigin="anonymous" referrerpolicy="no-referrer" />  
  <title>Twigg</title>
```

```

</head>
<body>
<!------- HEADER SECTION -->
<header class="header" >
  <a href="#" class="logo">
    
  </a>
  <nav class="navbar">
    <a href="/index.html" class="active">home</a>
    <a href="/about.html">about</a>
    <a href="/menu.html">menu</a>
    <a href="/products.html">products</a>
    <a href="/review.html">review</a>
    <a href="/contact.html">contact</a>
    <a href="/blog.html">blog</a>
  </nav>
  <div class="buttons">
    <button id="search-btn">
      <i class="fas fa-search"></i>
    </button>
    <button id="cart-btn">
      <i class="fas fa-shopping-cart"></i>
    </button>
    <button id="menu-btn">
      <i class="fas fa-bars"></i>
    </button>
  </div>
  <div class="search-form">
    <input type="text" class="search-input" id="search-box" placeholder="Search">
    <i class="fas fa-search"></i>
  </div>
  <div class="cart-items-container">
    <div class="cart-item">
      <i class="fas fa-times"></i>
      
      <div class="content">
        <h3>cart item 01</h3>
        <div class="price">Rs 50</div>
      </div>
    </div>
    <div class="cart-item">
      <i class="fas fa-times"></i>
      
      <div class="content">
        <h3>cart item 02</h3>
        <div class="price">Rs 100 </div>
      </div>
    </div>
    <div class="cart-item">
      <i class="fas fa-times"></i>
      
      <div class="content">
        <h3>cart item 03</h3>
        <div class="price">Rs 150 </div>
      </div>
    </div>
    <div class="cart-item">
      <i class="fas fa-times"></i>
      
      <div class="content">
        <h3>cart item 04</h3>

```

```
<div class="price">Rs 200 </div>  
</div>
```

```

        </div>
        <a href="#" class="btn">check out </a>
    </div>
</header>
<!-------HEADER SECTION -->

<!-------HOME SECTION -->
<section class="home" id="home">
    <div class="content">
        <h3>Twiggy</h3>
        <p>Where cravings meet convenience! Explore a world of flavors and order your favorite meals in just a few taps. Let's make your
hunger a thing of the past!</p>
        <a href="#" class="btn">order now</a>
    </div>
</section>
<!-------HOME SECTION -->

<!-------MENU SECTION -->
<section class="menu" id="menu">
    <h1 class="heading">our <span>menu</span></h1>
    <div class="box-container">
        <div class="box">

            <div class="box-head">
                
                <span class="menu-category">Pizza</span>
                <h3>6 Mini Pizzas</h3>
                <div class="price">Rs 349 <span>Rs 499</span></div>
            </div>
            <div class="box-bottom">
                <a href="#" class="btn">add to cart</a>
            </div>
        </div>
        <div class="box">

            <div class="box-head">
                
                <span class="menu-category">Burger</span>
                <h3>5 Mini Burgers</h3>
                <div class="price">Rs 249 <span>Rs 375</span></div>
            </div>
            <div class="box-bottom">
                <a href="#" class="btn">add to cart</a>
            </div>
        </div>
        <div class="box">

            <div class="box-head">
                
                <span class="menu-category">Pizza</span>
                <h3>2 Mixed Pizzas</h3>
                <div class="price">Rs 229 <span>Rs 299</span></div>
            </div>
            <div class="box-bottom">
                <a href="#" class="btn">add to cart</a>
            </div>
        </div>
        <div class="box">

            <div class="box-head">
                

```

Burger
<h3>3 Meatball Burgers</h3>


```
        <div class="price">Rs 299 <span>Rs 449</span></div>
    </div>
    <div class="box-bottom">
        <a href="#" class="btn">add to cart</a>
    </div>
</div>
</div>
</section>
<!-------MENU SECTION -->
```

```
<!-------PRODUCTS SECTION -->
<section class="products" id="products">
    <h1 class="heading">our <span>products</span> </h1>
    <div class="box-container">
        <div class="box">
            <div class="box-head">
                <span class="title">mini burger</span>
                <a href="#" class="name">Bacon Burger</a>
            </div>
            <div class="image">
                
            </div>
            <div class="box-bottom">
                <div class="info">
                    <b class="price">Rs 75</b>
                    <span class="amount">110gr / 300 Cal</span>
                </div>
                <div class="product-btn">
                    <a href="#">
                        <i class="fas fa-plus"></i>
                    </a>
                </div>
            </div>
        </div>
        <div class="box">
            <div class="box-head">
                <span class="title">cheese burger</span>
                <a href="#" class="name">cheese Burger</a>
            </div>
            <div class="image">
                
            </div>
            <div class="box-bottom">
                <div class="info">
                    <b class="price">Rs 120</b>
                    <span class="amount">140gr / 2500 Cal</span>
                </div>
                <div class="product-btn">
                    <a href="#">
                        <i class="fas fa-plus"></i>
                    </a>
                </div>
            </div>
        </div>
        <div class="box">
            <div class="box-head">
                <span class="title">Double burger</span>
                <a href="#" class="name">Double Burger</a>
            </div>
            <div class="image">
```

```

</div>
```

```

<div class="box-bottom">
  <div class="info">
    <b class="price">Rs 240</b>
    <span class="amount">440gr / 600 Cal</span>
  </div>
  <div class="product-btn">
    <a href="#">
      <i class="fas fa-plus"></i>
    </a>
  </div>
</div>
</div>
</div>
</div>
</section>
<!-------PRODUCTS SECTION -->

<!-------ABOUT US SECTION -->
<section class="about" id="about">
  <h1 class="heading">about <span>us</span> </h1>

  <div class="row">
    <div class="image">
      
    </div>
    <div class="content">
      <h3>What is the secret receipe of our burgers</h3>
      <div class="paragraph">
        <p>Our burgers are made with a special blend of freshly ground beef, seasoned to perfection with our secret spice mix. </p>
        <p>Each patty is grilled to juicy perfection and served on a toasted brioche bun. </p>
        <p>Topped with crisp lettuce, ripe tomatoes, melted cheese, and our signature sauce, it's a burger experience like no
other</p>
      </div>
      <a href="#" class="btn">Learn More</a>
    </div>
  </div>
</section>
<!-------ABOUT US SECTION -->

<!-------REVIEW SECTION -->
<section class="review" id="review">
  <h1 class="heading">customer's <span>review</span> </h1>
  <div class="box-container">
    <div class="box">
      
      <p> I'm always impressed by the range of restaurants available on this website. Whether I'm craving burgers, pizza, or Asian
cuisine, I can always find something to satisfy my appetite. The website is reliable, and the food quality is consistently excellent.</p>
      
      <h3>Yash Uskelwar</h3>
      <div class="stars">
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star-half-alt"></i>
      </div>
    </div>
    <div class="box">
      
      <p>The convenience of ordering food from this website is unmatched. I can order my favorite meals with just a

```

few clicks and have them delivered to my doorstep in no time. The website is well-designed, making it easy to navigate, and the delivery is always prompt.</p>

```


<h3>Harshita Dubey</h3>
<div class="stars">
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="fas fa-star-half-alt"></i>
</div>
</div>
<div class="box">
  
  <p>I love using this fast food delivery website! It's so easy to order my favorite meals from the comfort of my home. The
food always arrives hot and delicious, and the delivery is always on time.</p>
  
  <h3>Sneha Utekar</h3>
  <div class="stars">
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star-half-alt"></i>
  </div>
</div>
</div>
</div>
</section>
<section class="contact" id="contact">
  <h1 class="heading">contact <span>us</span> </h1>
  <div class="row">
    <iframe class="map"
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3771.8616688864765!2d72.88367541489297!3d19.0793845
87080392!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x3be7b96f91a6c81f%3A0x7c66614d7f8d15c9!2sKurla%2C%20M
umbai%2C%20Maharashtra!5e0!3m2!1sen!2sin!4v1658505945506!5m2!1sen!2sin" allowfullscreen="" loading="lazy" referrerpolicy="no-
referrer-when-downgrade"></iframe>

    <form>
      <h3>get in touch</h3>
      <div class="inputBox">
        <i class="fas fa-user"></i>
        <input type="text" placeholder="name">
      </div>
      <div class="inputBox">
        <i class="fas fa-envelope"></i>
        <input type="email" placeholder="email">
      </div>
      <div class="inputBox">
        <i class="fas fa-phone"></i>
        <input type="number" placeholder="number">
      </div>
      <input type="submit" class="btn" value="contact now">
    </form>
  </div>
</section>
<section class="blog" id="blog">
  <h1 class="heading">our <span>blog</span> </h1>
  <div class="box-container">
    <div class="box-full">
      <div class="image">
        
      </div>
      <div class="content">

```

[how to make burgers](#)
by admin / 15th Jan, 2024

A classic cheeseburger starts with a juicy beef patty seasoned with salt and pepper, grilled to perfection. Toast the burger buns until they're lightly golden and assemble the burger with a layer of crisp lettuce, a slice of ripe tomato, a few rings of onion, and a slice of your favorite cheese. Add a dollop of ketchup and mustard for that classic flavor combination. It's simple, delicious, and perfect for any burger lover.

```
</p>
<a href="#" class="btn">read more</a>
</div>
</div>
<div class="box-full">
  <div class="image">
    
  </div>
  <div class="content">
    <a href="#" class="title">how to make burgers</a>
    <span>by admin / 1st Feb, 2024</span>
    <p>For a BBQ bacon burger, start by grilling or frying your beef patty to your desired level of doneness. While the patty is cooking, fry up some crispy bacon. Toast your burger buns on the grill or in a pan until they're lightly golden. Assemble your burger with a generous spoonful of BBQ sauce on the bottom bun, followed by lettuce, tomato, onion, the beef patty, a slice of cheese, and the crispy bacon. Top it o with the top bun and enjoy the smoky, savory flavors of this delicious burger.</p>
    <a href="#" class="btn">read more</a>
  </div>
</div>
<div class="box-full">
  <div class="image">
    
  </div>
  <div class="content">
    <a href="#" class="title">how to make burgers</a>
    <span>by admin / 21st Feb, 2024</span>
    <p>For a spicy jalapeño burger, start by seasoning your beef patty with chili powder and cayenne pepper for an extra kick of heat. Grill or fry the patty until it's cooked to your liking. Slice some jalapeños and remove the seeds for a milder flavor, or leave them in for an extra spicy kick. Toast your burger buns until they're lightly golden and assemble your burger with a layer of lettuce, the beef patty, sliced jalapeños, a slice of pepper jack cheese, and a dollop of spicy mayo. It's a fiery and flavorful burger that's sure to please spice lovers.</p>
    <a href="#" class="btn">read more</a>
  </div>
</div>
</div>
</section>
<section class="footer">
  <div class="search">
    <input type="text" class="search-input" placeholder="Search">
    <button class="btn btn-primary">search</button>
  </div>
  <div class="share">
    <a href="#" class="fab fa-facebook"></a>
    <a href="#" class="fab fa-twitter"></a>
    <a href="#" class="fab fa-instagram"></a>
    <a href="#" class="fab fa-linkedin"></a>
    <a href="#" class="fab fa-pinterest"></a>
  </div>
  <div class="links">
    <a href="#home">home</a>
    <a href="#about">about</a>
    <a href="#menu">menu</a>
    <a href="#products">products</a>
    <a href="#review">review</a>
    <a href="#contact">contact</a>
    <a href="#blog">blog</a>
  </div>
```

<div class="credit">


```

        created by <span>Shubham Nakashe</span> | all rights reserved!
    </div>
</section>
<!-------FOOTER SECTION -->

<script src="./script.js"></script>
<script src="./app.js"></script>
</body>
</html>

```

2) app.js

```

if ('serviceWorker' in navigator) {
    window.addEventListener('load', () => {
        navigator.serviceWorker.register('/service-worker.js')
            .then(registration => {
                console.log('Service Worker registered with scope:', registration.scope);
            })
            .catch(error => {
                console.error('Service Worker registration failed:', error);
            });
    });
}

```

3) service-worker.js

```

var cacheName = "Twiggy";
const assetsToCache = [
    '/',
    '/index.html',
    '/style.css',
    '/app.js',
    '/blog.html',
    '/about.html',
    '/contact.html',
    '/menu.html',
    '/products.html',
    '/review.html',
    '/script.js'
];

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

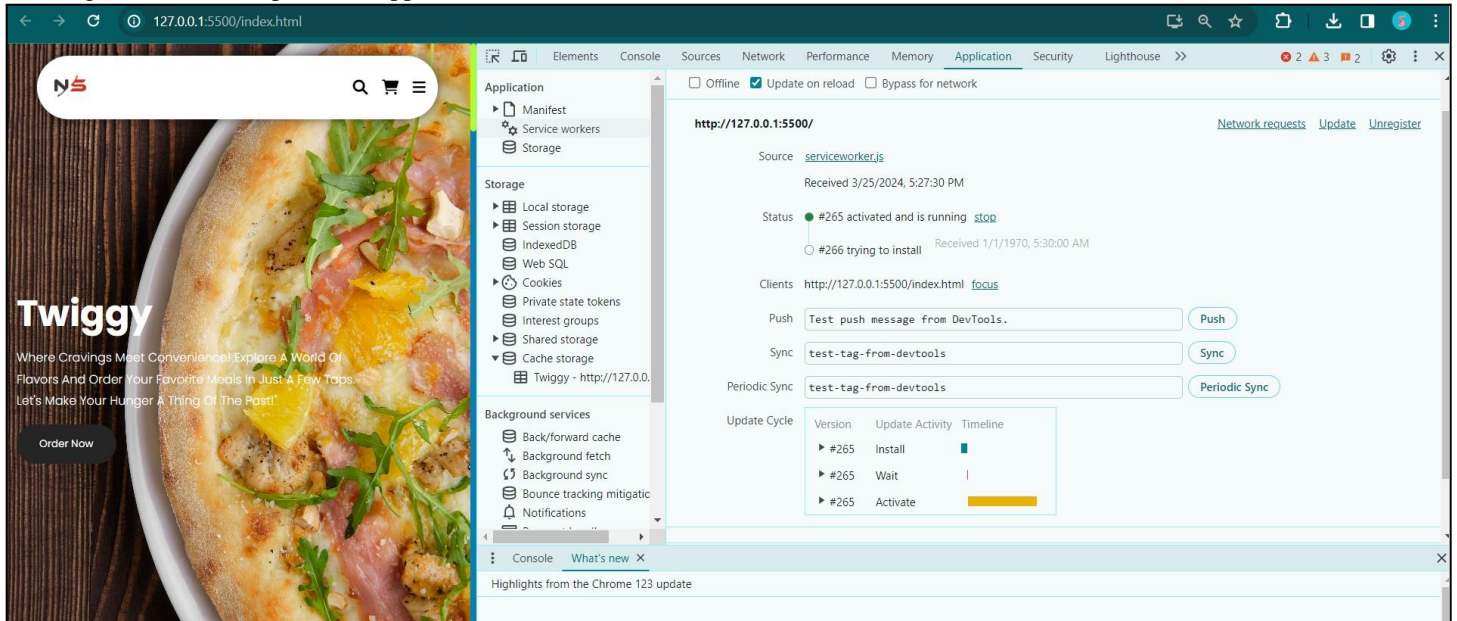
self.addEventListener('activate', event => {
    event.waitUntil(
        caches.keys().then(cacheNames => {
            return Promise.all(
                cacheNames.filter(name => {
                    return name !== cacheName;
                })
            );
        })
    );
});

```

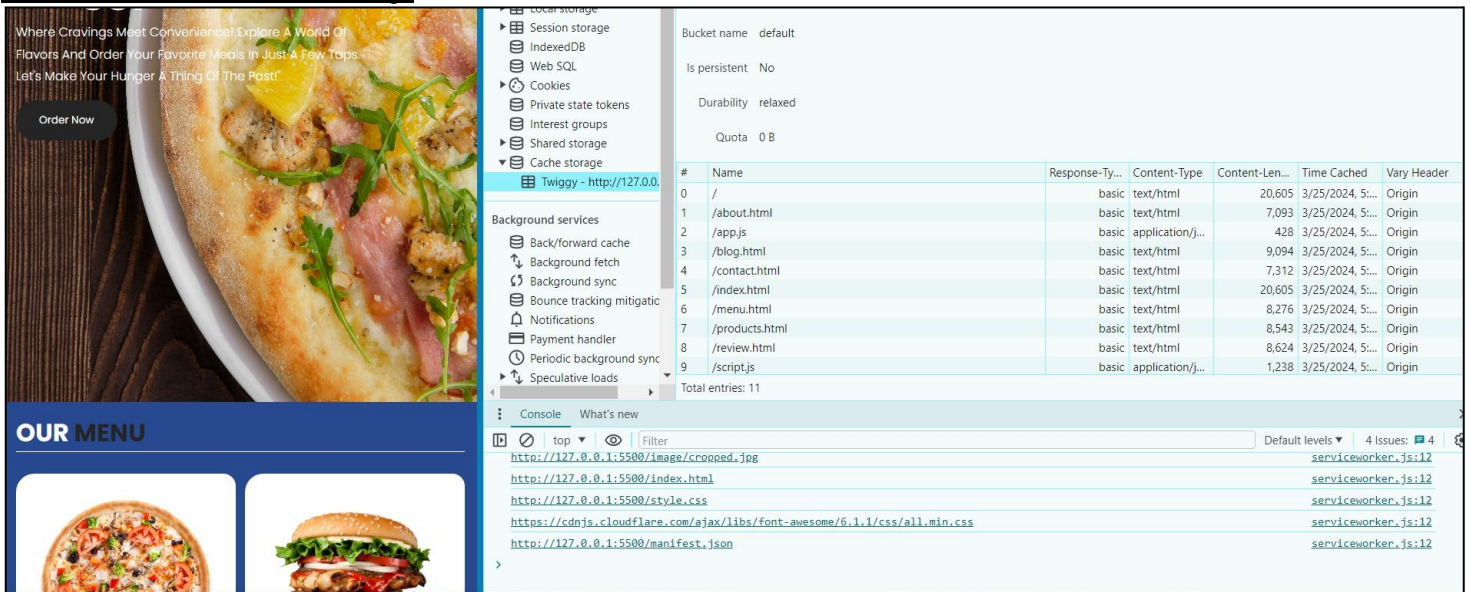
```
).map(name => {  
  return caches.delete(name);  
})
```

```
);  
})  
);  
});
```

Now Right click —> Inspect —> Application —> Service workers



Now scroll down to Cache Storage



Conclusion:

- Registered a service worker, and completed the installation and activation process for a new service worker.
- Also checked the Cache Storage