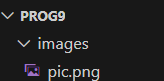
LAB9

9 Demonstrate online and offline webpage load using pwa,service worker and caching

Step 1: Place all the icons in the assets directory

Note: Icons must be square and greater than or equal to 144x144px



Step 2: Create the manifest file

manifest.json

{

"name": "Fetch Github Details",

"short\_name": "API Fetch",

"display": "standalone",

"start\_url": "/",

"id": "/",

"orientation": "portrait-primary",

"description": "Get user details from Github using its API",

"icons": [

{

"src": "images/pic.png",

"type": "image/png",

"sizes": "256x256"

}

],

"background\_color": "#AA0000",

"theme\_color": "#CC0000"

}

**Step 3:** Create the service worker file

sw.js

self.addEventListener("install", (e) => {

console.log("installed");

});

self.addEventListener("activate", (e) => {

console.log("activated");

}):

self.addEventListener("fetch", (e) => {

console.log("fetched");

// Fetching user data from the GitHub API

fetch("https://api.github.com/users/Jayasimhasr")

.then((response) => {

return response.json();

})

.then((json) => {

console.log(json);

})

.catch((error) => {

console.log(error);

});

});

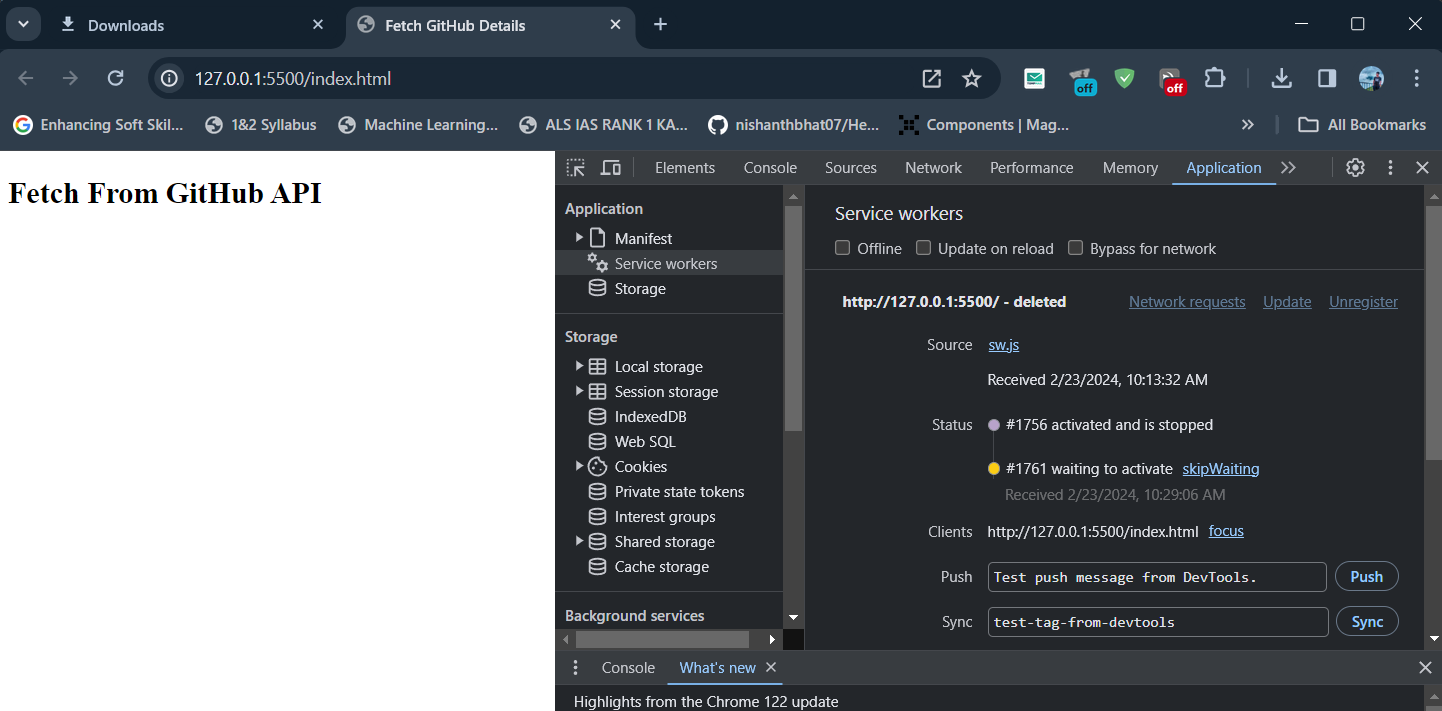
**Step 4:** Create a html page to register the service worker

index.html

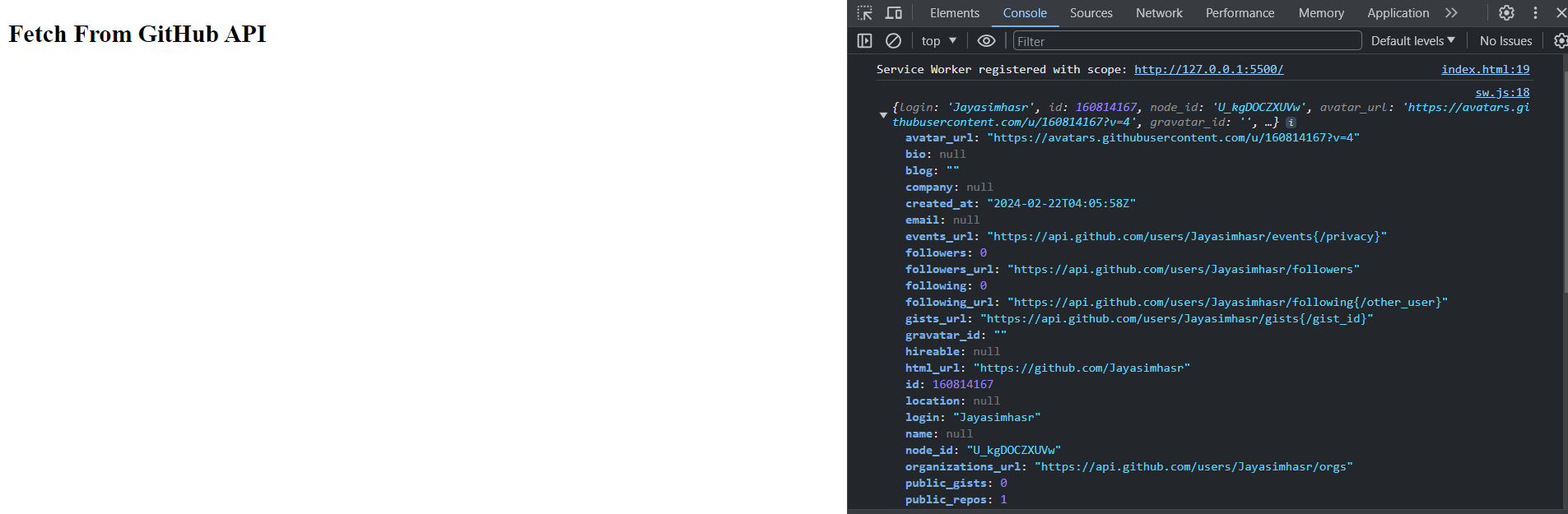
**Students can use**

|  |
| --- |
| **<!DOCTYPE html>**  **<html>**  **<head>**  **</head>**  **<link rel="manifest" href="manifest.json">**  **<body>**  **<h1>Github Profile</h1>**  **<input type="text" id="text\_input">**  **<input type="button" onclick="gitdata()" value="Get Profile">**  **<div id='git\_image'>**  **<img src="" alt="profile image" id="image">**  **<p id='para'></p>**  **</div>**  **</body>**  **<script>**  **function gitdata(){**  **var url = "https://api.github.com/users/" + document.getElementById("text\_input").value;**  **fetch(url)**  **.then(response => {**  **if (!response.ok) {**  **throw new Error('Network response was not ok');**  **}**  **return response.json();**  **})**  **.then(data => {**  **console.log(data.avatar\_url);**  **var im = document.getElementById("image");**  **im.setAttribute("src", data.avatar\_url);**  **var gi = document.getElementById("para");**  **gi.innerHTML = JSON.stringify(data);**  **})**  **.catch(error => {**  **console.error('There was a problem with the fetch operation:', error);**  **});**  **}**  **</script>**  **</html>** |

**Output- (make sure to unregister the service worker from the previous programs in the applications tab of the inspect menu)**



Online support of webpage Display through service worker and caching



Copy the displayed URL and open through another Browser

