Data.json

[

{

"company": "3M",

"description": "3M, based in Minnesota, may be best known for its Scotch tape and Post-It Notes, but it also produces sand paper, adhesives, medical products, computer screen filters, food safety items, stationery products and many products used in automotive, marine, and aircraft industries.",

"initial\_price": 44.28,

"price\_2002": 56.27,

"price\_2007": 95.85,

"symbol": "MMM"

},

{

"company": "Amazon.com",

"description": "Amazon.com, Inc. is an online retailer in North America and internationally. The company serves consumers through its retail Web sites and focuses on selection, price, and convenience. It also offers programs that enable sellers to sell their products on its Web sites, and their own branded Web sites. In addition, the company serves developer customers through Amazon Web Services, which provides access to technology infrastructure that developers can use to enable virtually various type of business. Further, it manufactures and sells the Kindle e-reader. Founded in 1994 and headquartered in Seattle, Washington.",

"initial\_price": 89.38,

"price\_2002": 17.01,

"price\_2007": 93.43,

"symbol": "AMZN"

}

]

Index.html

<!DOCTYPE html>

<html>

<head>

<title>Stock Display</title>

<link rel="manifest" href="manifest.json"> <style>

table{

border-collapse: collapse;

}

td, th {

padding: 1em;

border: 1px solid black;

}

</style>

</head>

<body>

<center>

<h2>Stock Market Data Display</h2>

<div id="stock"></div>

</center>

<script>

if('serviceWorker' in navigator) {

navigator.serviceWorker.register("sw.js")

.then((e) => {

if(e.active === null){

console.log("registered");

}

});

} else {

console.log("browser does not support service worker");

}

// data from https://gist.github.com/tanveery/4ac939d2ad27954da4c8db13e10ef7bd

fetch("data.json")

.then((res) => {

return res.json();

})

.then((stock) => {

// creating the table for displaying the stock data var table = "<table>";

table +="<tr><th>"+Company+"</th><th>"+Description+"</th><th>"+Initial\_Price+"</th><th>"+Price\_in\_2007+"</th><th>"+Symbol+"</th></tr>";

for(item of stock) {

table += "<tr><td>"+item.company+"</td><td>"+item.description+"</td><td>"+item.initial\_price+"</td><td>"+item.price\_2007+"</td><td>"+item.symbol+"</td></tr>"

;

}

table += "</table>";

// inserting the table into html

document.getElementById("stock").innerHTML = table;

})

</script>

</body>

</html>

Manifest.json

{

"name": "Stock Display",

"short\_name": "Stock-Display",

"display": "standalone",

"start\_url": "/",

"id": "/",

"description": "Display stock from json file",

"icons": [

{

"src": "assets/rv.png",

"type": "image/png",

"sizes": "512x512",

"purpose": "maskable"

},

{

"src": "assets/rv1.png",

"type": "image/png",

"sizes": "512x512",

"purpose": "any"

}

],

"background\_color": "#00AA00",

"theme\_color": "#00CC00"

}

Sw.js

self.addEventListener("install", (e) => { console.log("installed");

// creating a cache storage and adding all the files required for the web page to run even when it’s offline

caches.open("stock").then((cache) => { cache.add("/"); cache.add("./assets/rv.png"); cache.add("./assets/rv1.png"); cache.add("./data.json"); cache.add("./index.html"); cache.add("./manifest.json"); cache.add("./sw.js");

})

.catch((err) => { console.log(err);

});

})

self.addEventListener("activate", (e) => { console.log("activated");

})

self.addEventListener("fetch", (e) => { console.log("fetched");

// intercepting every fetch request and responding from the cache when it’s offline

e.respondWith(

// checking if requested data is present in the cache

caches.match(e.request)

.then((res) => {

// respond from cache when fetch function doesn’t return any response(i.e, the system is offline)

return res || fetch(e.request);

})

.catch((err) => { console.log(err);

})

);

})