Caching individual pages for offline use

Caching individual pages by hand

Most of the examples use a hand-coded service worker with a customized version of the registration event

```
// This registers the service worker and sets
// up the files to cache for the particular
// page.
if ('serviceWorker' in navigator) {
  'serviceWorker'// Attempt to register itrviceWorker.register('/sw.js').
    // Success
    '/sw.js'// Success
    console.log('ServiceWorker registration successful');
  }).catch(function(err) {
    // Fail
    console.log('ServiceWorker registration failed: ', err);
  });

  // Set variables for use in the event listenerndow.location.pathname;
  var cacheButton = document.querySelector('.offline-btn');

  // Event listener
  if('.offline-btn'// Event listeneraddEventListener('click', function(eve
        event.preventDefault();

        const urlsToCache = [
          currentPath,
          ...performance.getEntriesByType('resource').map((r) => r.name),
        ];

      // Open the unique cache for this URL'click'// Open the unique cache
      caches.open('offline-' + currentPath).then(function(cache) {
        var updateCache = cache.addAll(urlsToCache);
        // Update UI to indicate success
        // Or catch any errors if it doesn't succeed     console.log('Art
        }).catch(function (error) {
          console.log('Article could not be saved offline.');
        });
      });
    });
  }
}
'Article is now available offline.'
```

Links and Resources

- Implementing “Save For Offline” with Service Workers
- Service Worker Cache Script
- Jake Archibald’s Offline Cookbook
- Mozilla Service Worker Cookbook
- CSS Tricks Service Worker for Offline