IN4MATX 285:
Interactive Technology Studio

Programming: Browser Extensions

Today’s goals

By the end of today, you should be able to…

- Articulate the capabilities and limits of browser extensions
- Describe and implement the structure of a Chrome extension
- Create a simple Chrome extension that modifies the DOM

What does a browser extension do?

What does a browser extension do?

- Edits how a website is rendered in a browser
- Adds functionality beyond what the website developer envisioned

One example: Boomerang

Boomerang

- Very early (~2013) browser extension on top of Gmail
- Filled in usability gaps in Gmail
  - Enabled schedule-sending emails, recurring emails, snoozing, etc.
- Google has since added many of these features
  - Schedule-send in 2019

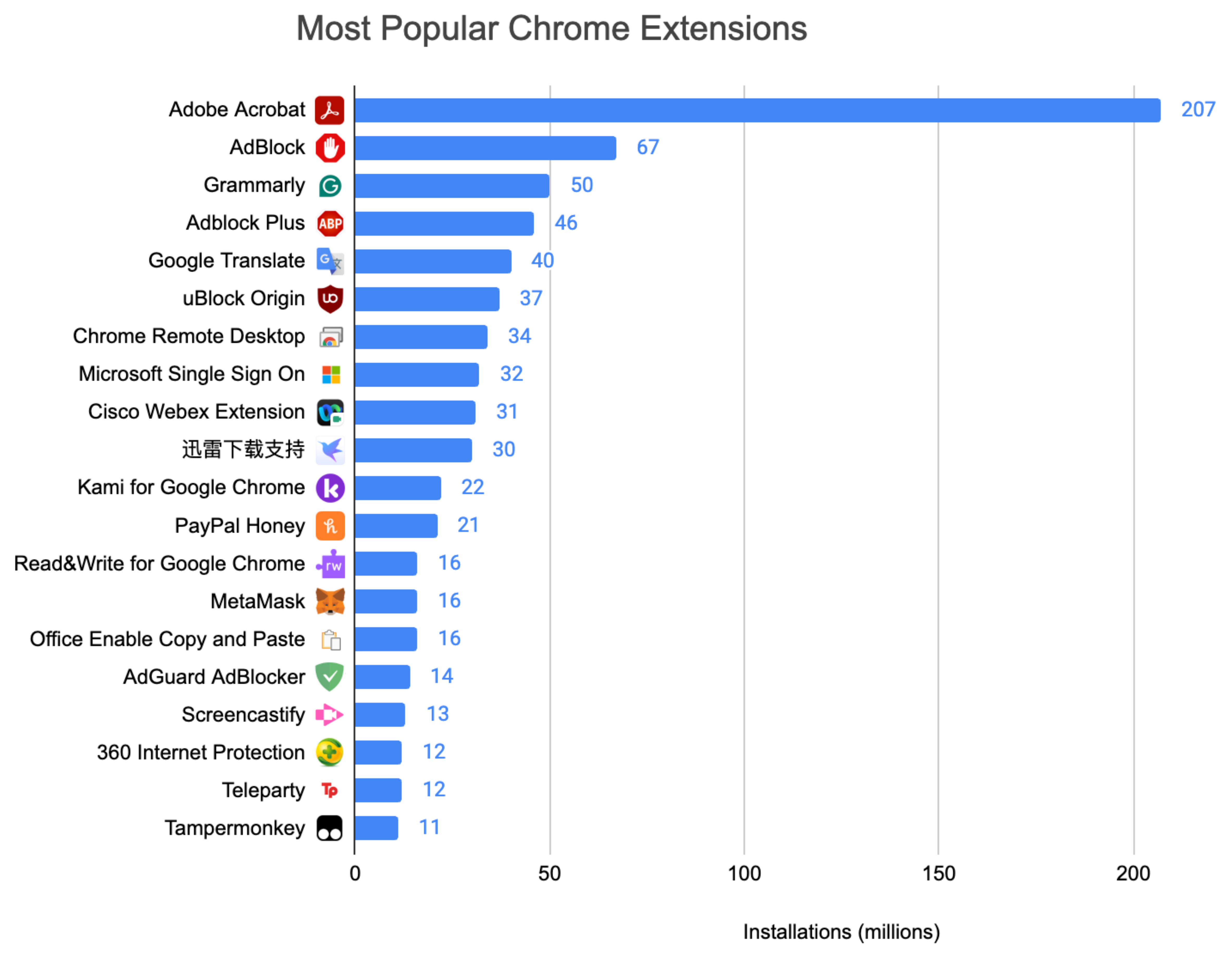
Browser extensions: why bother?

Browser extensions: why bother?

- Improve on existing web interfaces
    - Usability
    - Accessibility
- Add entirely new features
    - Rendering PDFs
    - Ad blocking
    - Translation

Browser extensions: why bother?

- Broad categories
  - Ad blocking
  - PDF rendering
  - Collaboration and screen sharing



https://www.debugbear.com/blog/chrome-extension-statistics

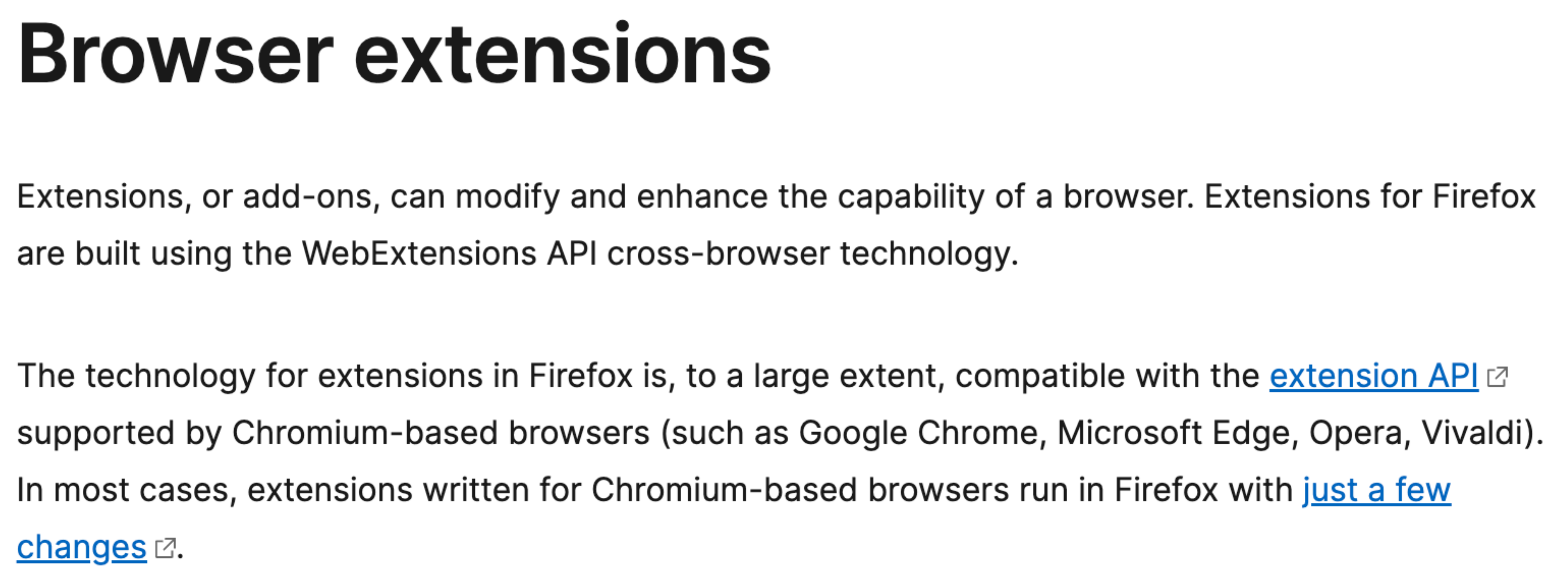
Browser extensions: weaknesses

- Extensions can only edit what is being shown on the *client*
- But, extensions typically have a lot of access to what you're doing in your *client*, and can send that information to *servers*
  - Browsing history
  - User input
- Therefore, it's important to have trust in the extensions you install

Making a browser extension

Making a browser extension

- As we've discussed, browsers implement JavaScript slightly differently
  - A "WebExtensions" standard exists, but isn't uniformly followed
- Browser extensions therefore vary as well
  - Chrome, Firefox, and Safari extensions are all slightly different
- We'll focus on Chrome



Browser extensions

Extensions, or add-ons, can modify and enhance the capability of a browser. Extensions for Firefox are built using the WebExtensions API cross-browser technology.

The technology for extensions in Firefox is, to a large extent, compatible with the extension API supported by Chromium-based browsers (such as Google Chrome, Microsoft Edge, Opera, Vivaldi). In most cases, extensions written for Chromium-based browsers run in Firefox with just a few changes.

https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions

Making a Chrome extension

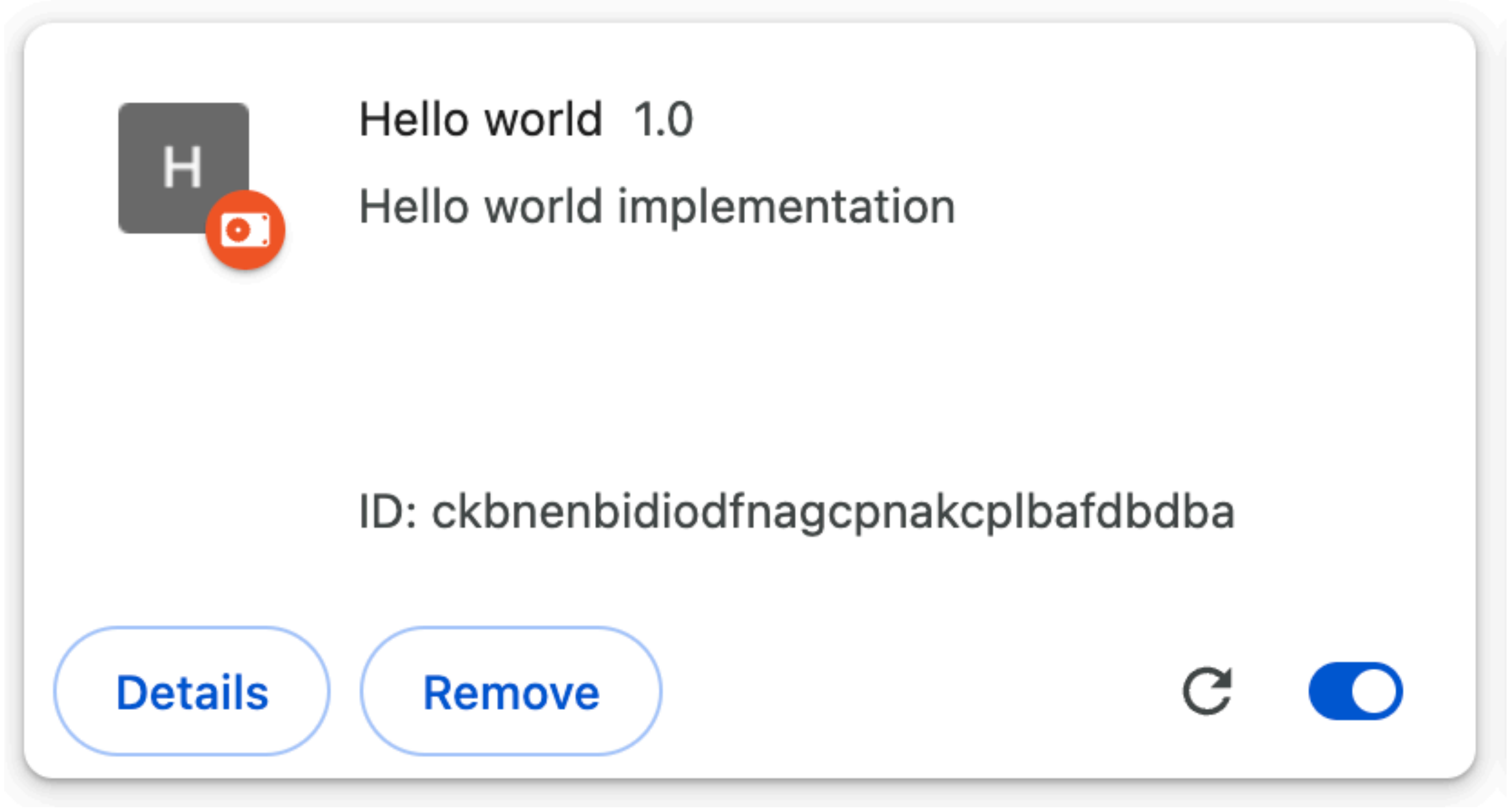
manifest.json

- All extensions have a `manifest.json` file which provides details on structure and permissions of the extension
- json: Javascript object
- Manifest often points to other files (javascript, CSS, images) which the script uses

manifest.json

- Name, description, version, manifest version
  - Manifest 3 is the latest, better privacy protections

```
{
   "name": "Hello world",
   "description": "Hello
world implementation",
   "version": "1.0",
   "manifest_version": 3
}
```


Hello world 1.0

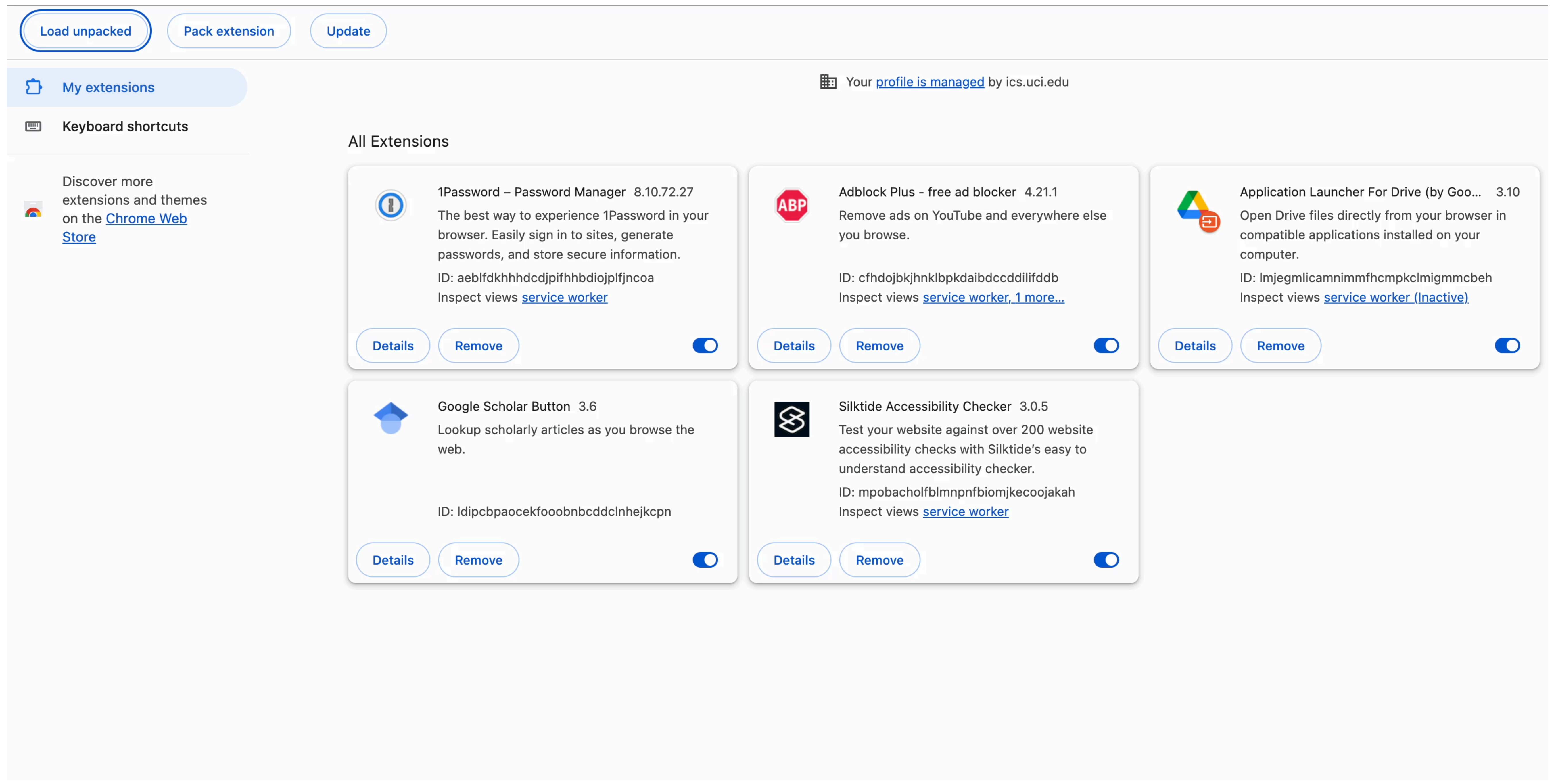
Hello world implementation

ID: ckbnenbidiodfnagcpnakcplbafdbdba

https://developer.chrome.com/docs/extensions/develop/migrate/what-is-mv3

Installing an extension

- chrome://extensions/
- Turn on "developer mode" (top-right) Developer mode
- "Load unpacked"
    - Folder with manifest.json and other extension files



Discover more extensions and themes on the Chrome Web Store

Your profile is managed by ics.uci.edu

All Extensions

1Password – Password Manager 8.10.72.27
The best way to experience 1Password in your browser. Easily sign in to sites, generate passwords, and store secure information.
ID: aeblfdkhhhdcdjpifhhbdiojplfjncoa
Inspect views service worker

Adblock Plus - free ad blocker 4.21.1
Remove ads on YouTube and everywhere else you browse.
ID: cfhdojbkjhnklbpkdaibdccddilifddb
Inspect views service worker, 1 more...

Application Launcher For Drive (by Goo... 3.10
Open Drive files directly from your browser in compatible applications installed on your computer.
ID: lmjegmlicamnimmfhcmpkclmigmmcbeh
Inspect views service worker (Inactive)

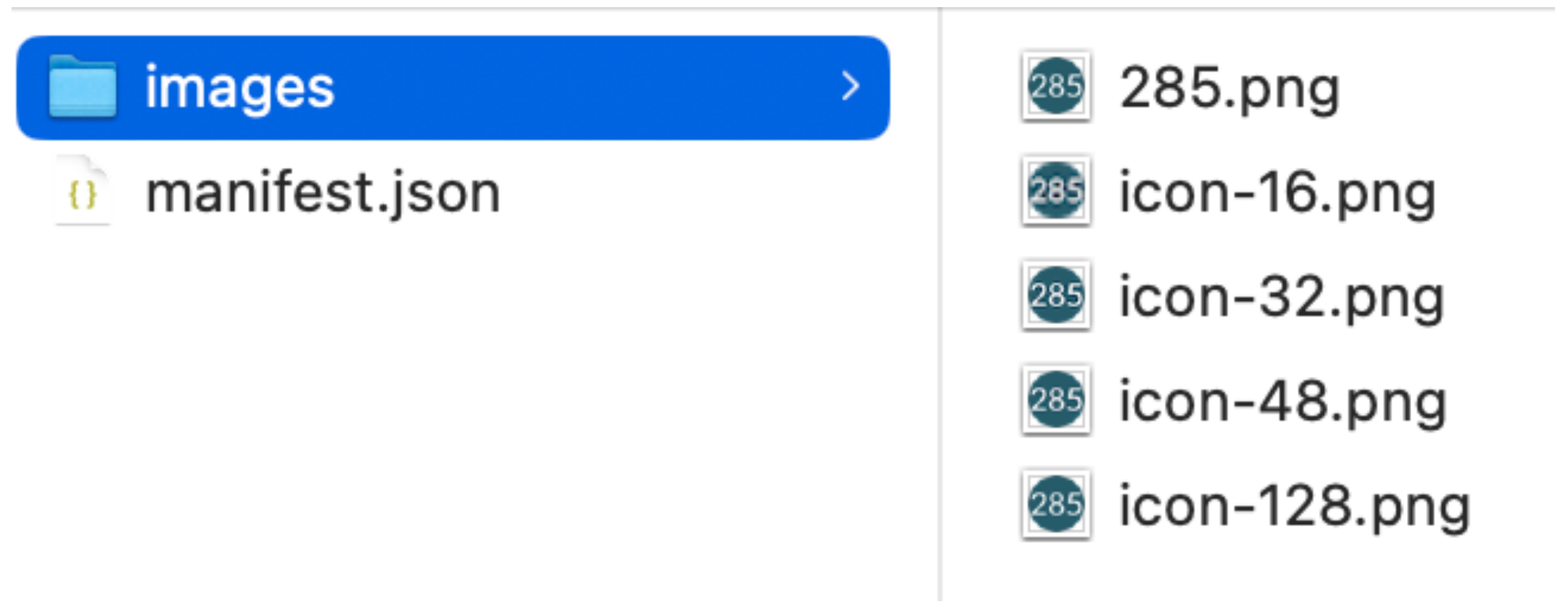
Google Scholar Button 3.6
Lookup scholarly articles as you browse the web.
ID: ldipcbpaocekfooobnbcddclnhejkcpn

Silktide Accessibility Checker 3.0.5
Test your website against over 200 website accessibility checks with Silktide's easy to understand accessibility checker.
ID: mpobacholfblmnpnfbiomjkecoojakah
Inspect views service worker

Icons

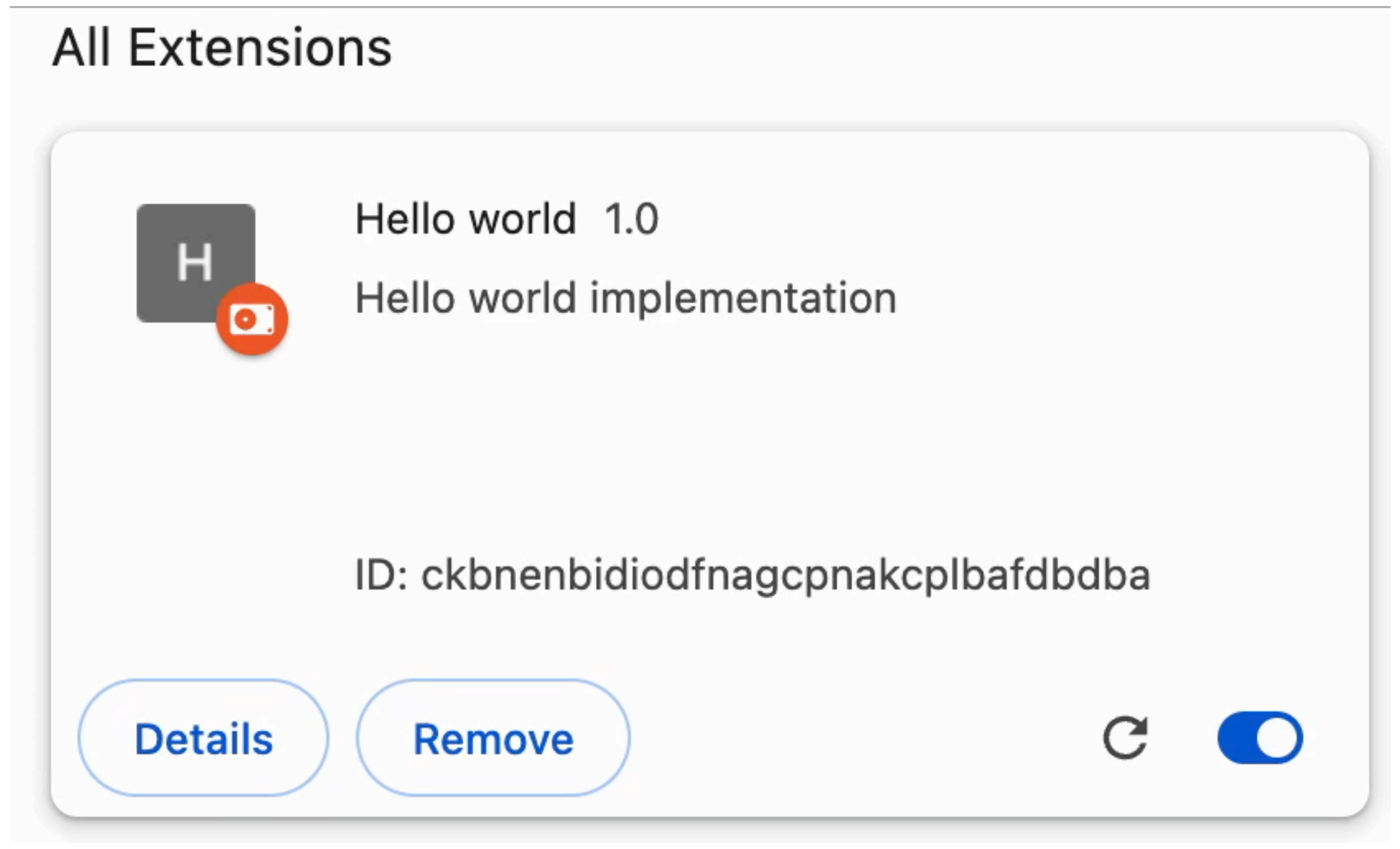
- Provide differently-sized icons for the browser bar and extensions page
- Add relative URLs to the `manifest.json`

```
{
  "name": "Hello world",
  "description": "Hello world implementation",
  "version": "1.0",
  "manifest_version": 3,
  "icons": {
        "16": "images/icon-16.png",
        "32": "images/icon-32.png",
        "48": "images/icon-48.png",
        "128": "images/icon-128.png"
  }
}
```



Refreshing an extension

- Need to *refresh* the extension to see all changes
- For example, to see the icons
  - But also when you change the underlying code


All Extensions

Hello world 1.0

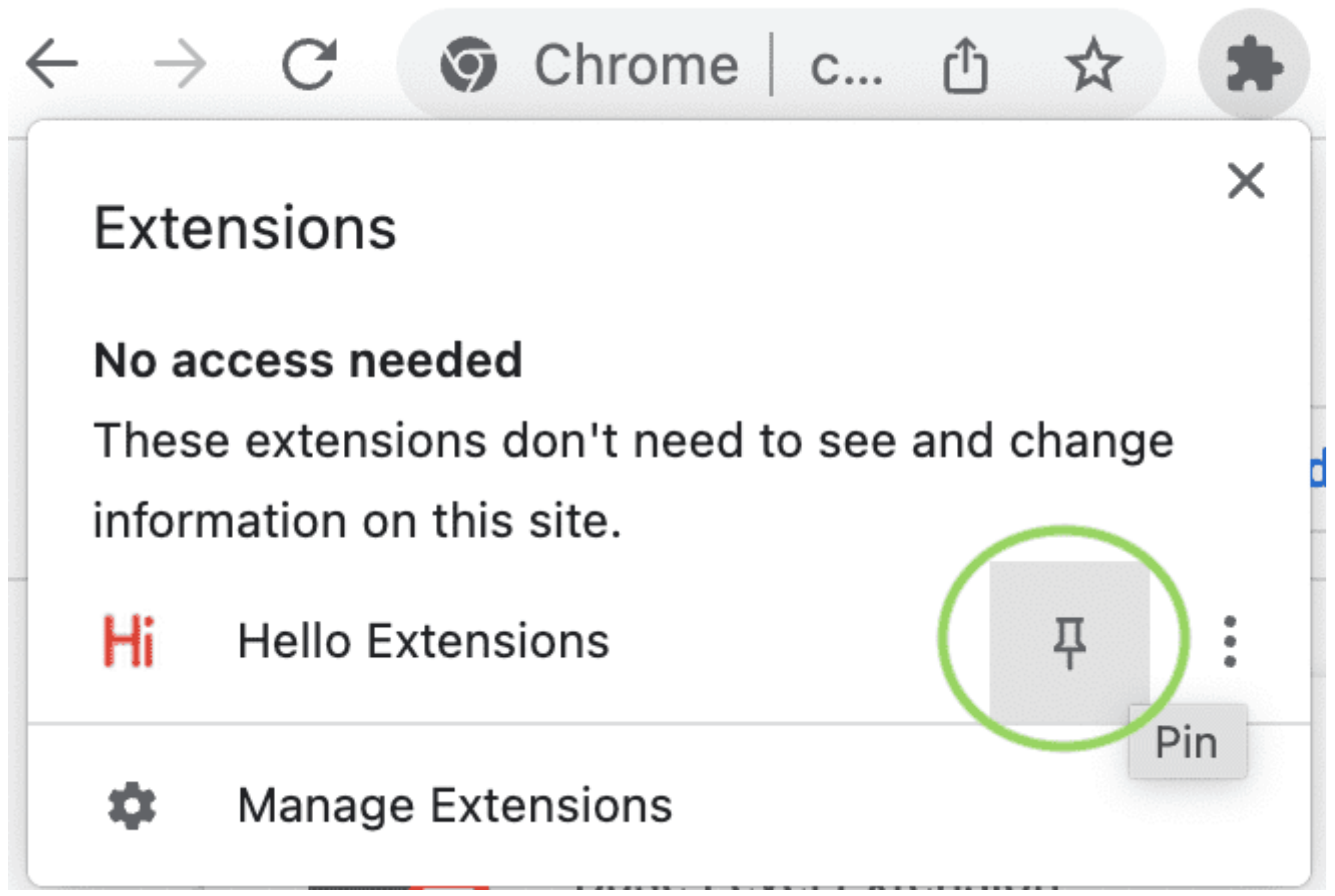
Hello world implementation

ID: ckbnenbidiodfnagcpnakcplbafdbdba

Extension component	Requires extension reload
The manifest	Yes
Service worker	Yes
Content scripts	Yes (plus the host page)
The popup	No
Options page	No
Other extension HTML pages	No

Pinning an extension

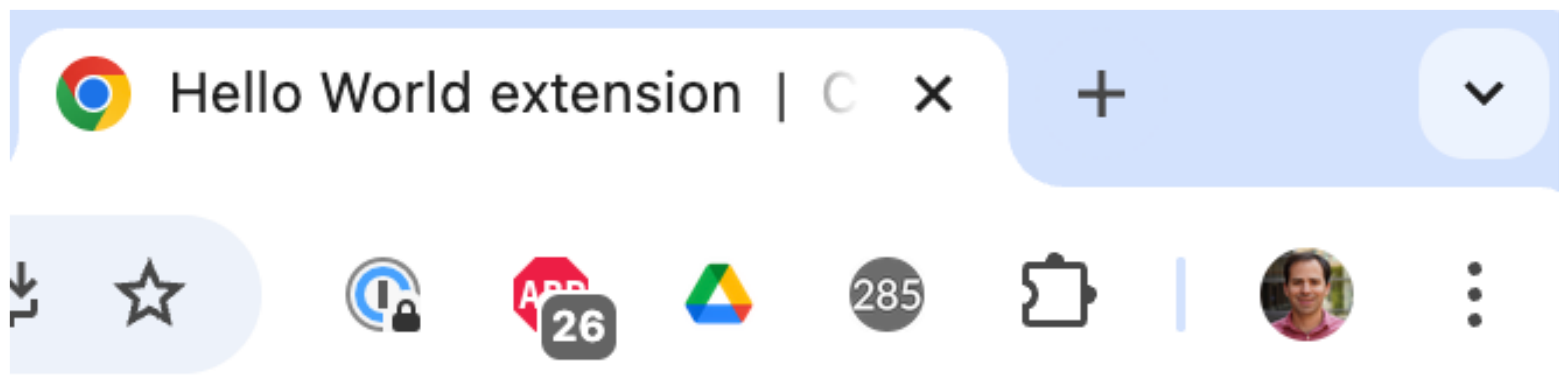
- Extensions are listed in the extensions bar (puzzle piece)
- It's often helpful to "pin" the extension you're working on



Extensions

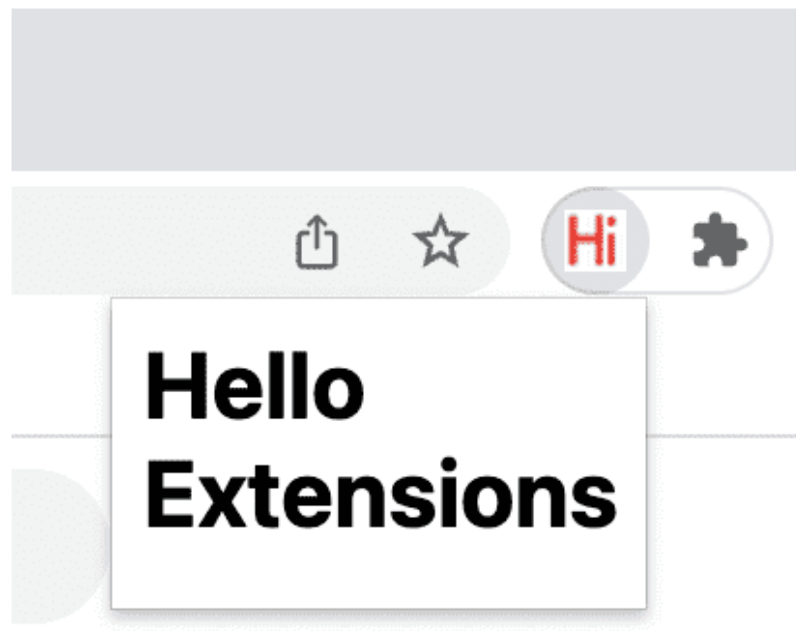
No access needed

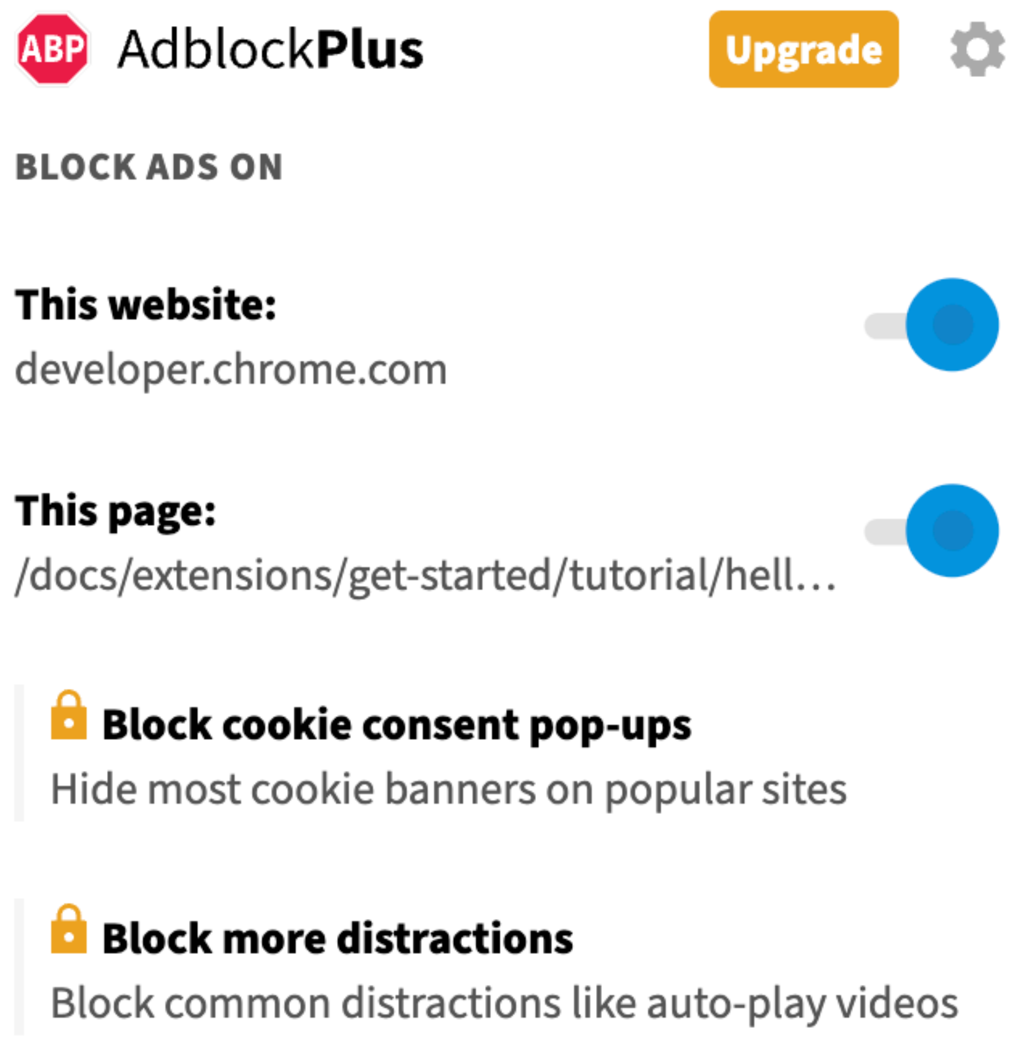
These extensions don't need to see and change information on this site.



Extension capabilities

- Extensions can provide lots of different capabilities
- New HTML pages and popups
  - Useful for menus and sharing information
- Service workers
  - Useful for background processing of page information
- We'll focus on *content scripts*
  - New Javascript and CSS files which modify the DOM of the current page




AdblockPlus Upgrade

BLOCK ADS ON

This website: developer.chrome.com

This page: /docs/extensions/get-started/tutorial/hell…

Block cookie consent pop-ups
Hide most cookie banners on popular sites

Block more distractions
Block common distractions like auto-play videos

Content scripts

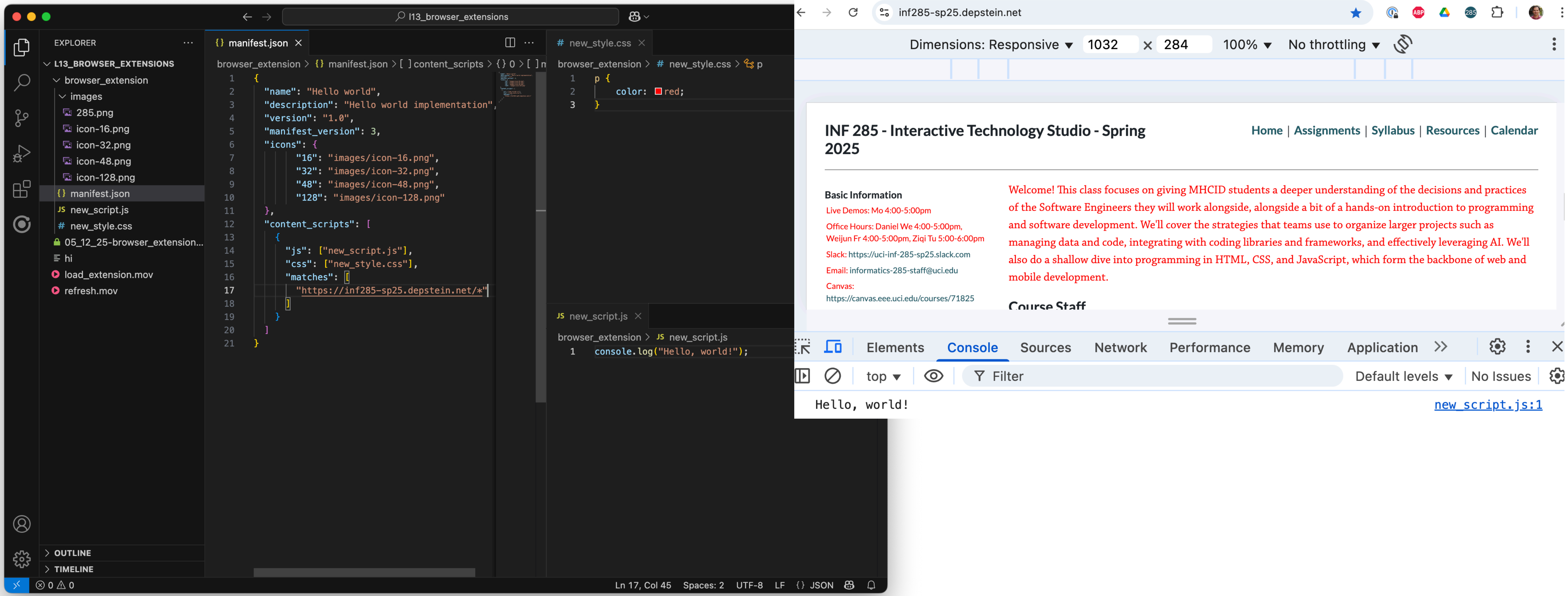
- Javascript and/or CSS files which get loaded every time you open a particular URL
- Add a `content_scripts` array to `manifest.json` with js and css file(s), and what URL to match
- Additional options for more sophisticated URL matching and when to load, which we'll mostly ignore

```
"content_scripts": [ {
  "js": ["new_script.js"],
  "css": ["new_style.css"],
  "matches": [ “https://inf285-sp25.depstein.net/*”]
} ]
```

https://developer.chrome.com/docs/extensions/reference/manifest/content-scripts

Hello world

- Putting it all together…



manifest.json

```json
{
  "name": "Hello world",
  "description": "Hello world implementation",
  "version": "1.0",
  "manifest_version": 3,
  "icons": {
      "16": "images/icon-16.png",
      "32": "images/icon-32.png",
      "48": "images/icon-48.png",
      "128": "images/icon-128.png"
  },
  "content_scripts": [
    {
      "js": ["new_script.js"],
      "css": ["new_style.css"],
      "matches": [
        "https://inf285-sp25.depstein.net/*"
      ]
    }
  ]
}
```

new_style.css

```css
p {
    color: red;
}
```

new_script.js

```javascript
console.log("Hello, world!");
```

Ln 17, Col 45   Spaces: 2   UTF-8   LF   { } JSON

inf285-sp25.depstein.net

Dimensions: Responsive   1032 × 284   100%   No throttling

INF 285 - Interactive Technology Studio - Spring 2025

Home | Assignments | Syllabus | Resources | Calendar

Basic Information

Live Demos: Mo 4:00-5:00pm

Office Hours: Daniel We 4:00-5:00pm, Weijun Fr 4:00-5:00pm, Ziqi Tu 5:00-6:00pm

Slack: https://uci-inf-285-sp25.slack.com

Email: informatics-285-staff@uci.edu

Canvas: https://canvas.eee.uci.edu/courses/71825

Welcome! This class focuses on giving MHCID students a deeper understanding of the decisions and practices of the Software Engineers they will work alongside, alongside a bit of a hands-on introduction to programming and software development. We'll cover the strategies that teams use to organize larger projects such as managing data and code, integrating with coding libraries and frameworks, and effectively leveraging AI. We'll also do a shallow dive into programming in HTML, CSS, and JavaScript, which form the backbone of web and mobile development.

Course Staff

```
Hello, world!                                   new_script.js:1
```

Web accessible resources

- Can make resources like images available in extensions

```
"web_accessible_resources": [
  {
    "resources": ["images/285.png"],
    "matches": ["https://inf285-sp25.depstein.net/*"]
  }
]
```

- Resources get encoded, can be accessed with `chrome.runtime.getURL`

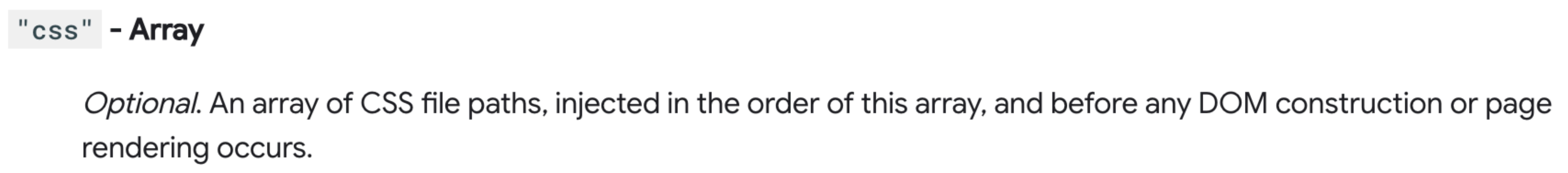
```
let images = document.getElementsByTagName('img');
for(let image of images) {
    image.src = chrome.runtime.getURL("images/285.png");
}
```

https://developer.chrome.com/docs/extensions/reference/manifest/web-accessible-resources

CSS priority

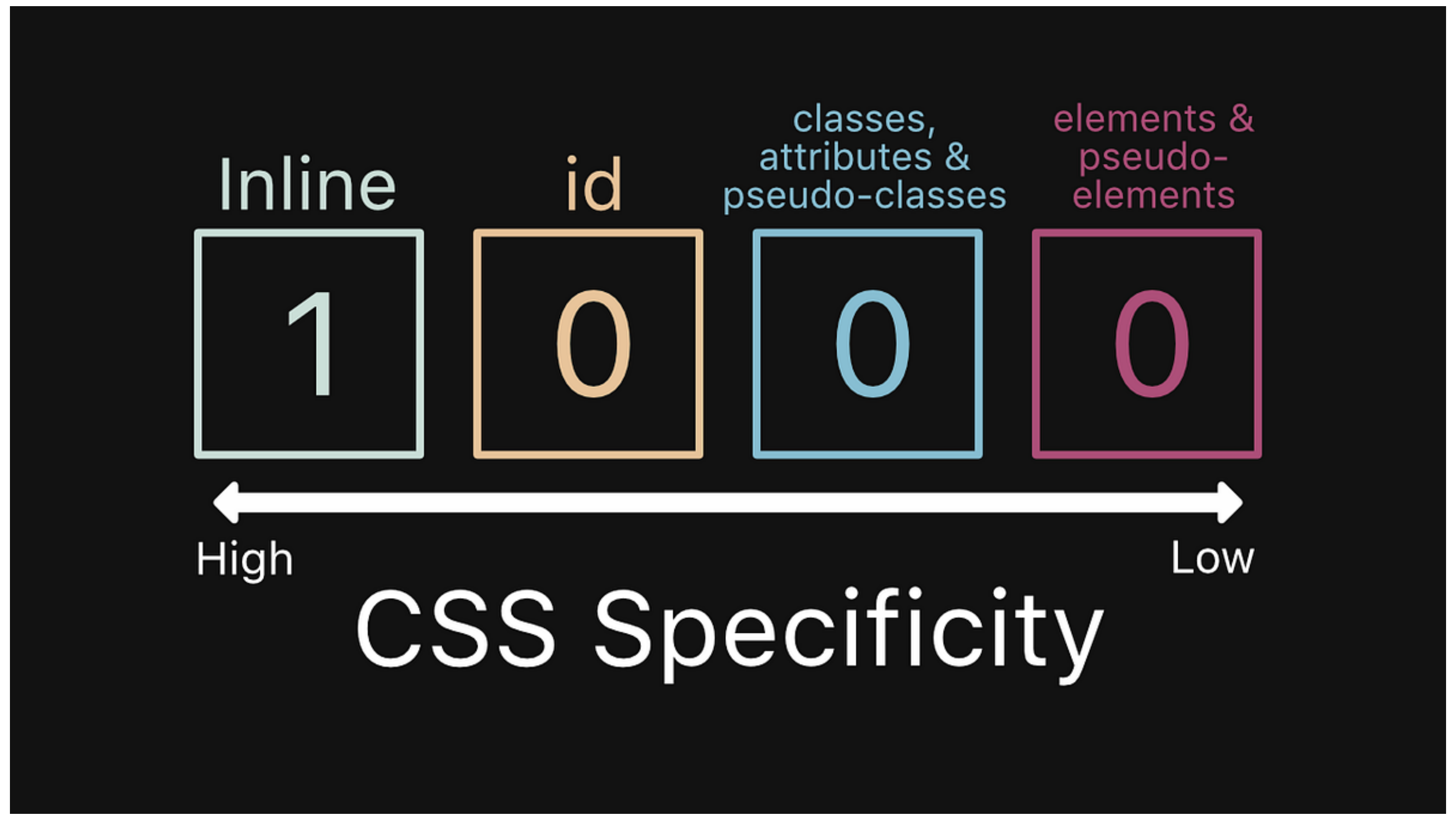
- CSS content scripts get injected *before* any CSS loaded by the page itself

- This means you need to use a more specific selector than the page if you want your script to apply
  - Or use `!important` (bad form)
  - Or use Javascript to modify, since that will create inline styles



`"css"` - **Array**

Optional. An array of CSS file paths, injected in the order of this array, and before any DOM construction or page rendering occurs.


Inline
id
classes, attributes & pseudo-classes
elements & pseudo-elements
1
0
0
0
High
Low
CSS Specificity

https://developer.chrome.com/docs/extensions/reference/manifest/content-scripts

Reflecting on extensions

- Extensions are powerful!
  - You can overwrite the DOM to create more usable and accessible interactions
  - You can add entirely new features
  - There's a ton of features beyond what I described today
- Extensions are dangerous
  - They can read and write a lot of what you're doing on the web
- Keep scope small by only allowing access to particular pages and functionality

Today’s goals

By the end of today, you should be able to…

- Articulate the capabilities and limits of browser extensions
- Describe and implement the structure of a Chrome extension
- Create a simple Chrome extension that modifies the DOM

IN4MATX 285:
Interactive Technology Studio

Programming: Browser
Extensions