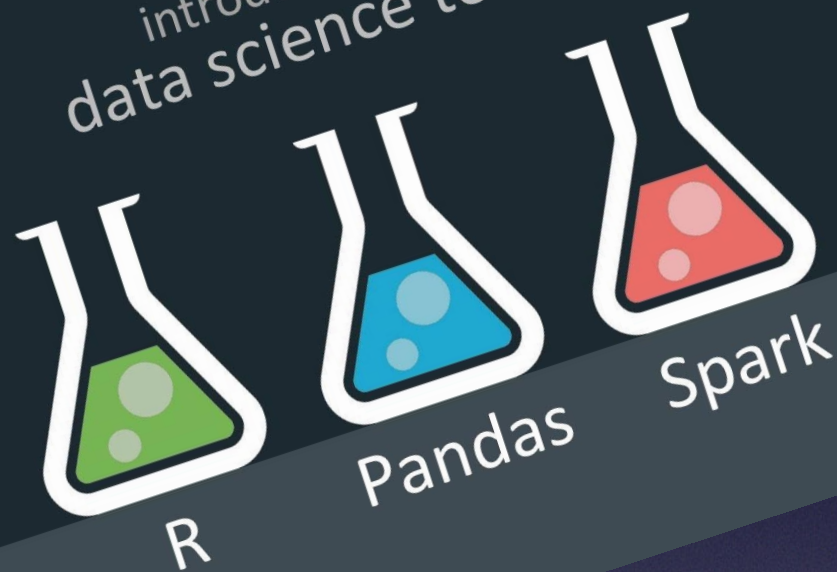


How to Develop Chrome Extension

Vector Li

introduction to
data science tools



Building Combat Support in the Brigade Engineer Battalion: *Integrating the Capabilities of Signal Companies*

Learn from Peers Presents

Pretty Fly for a Brown Guy

Arjun's Plane and Simple Guide to
Getting a Private Pilot's License

Colocating Command Posts

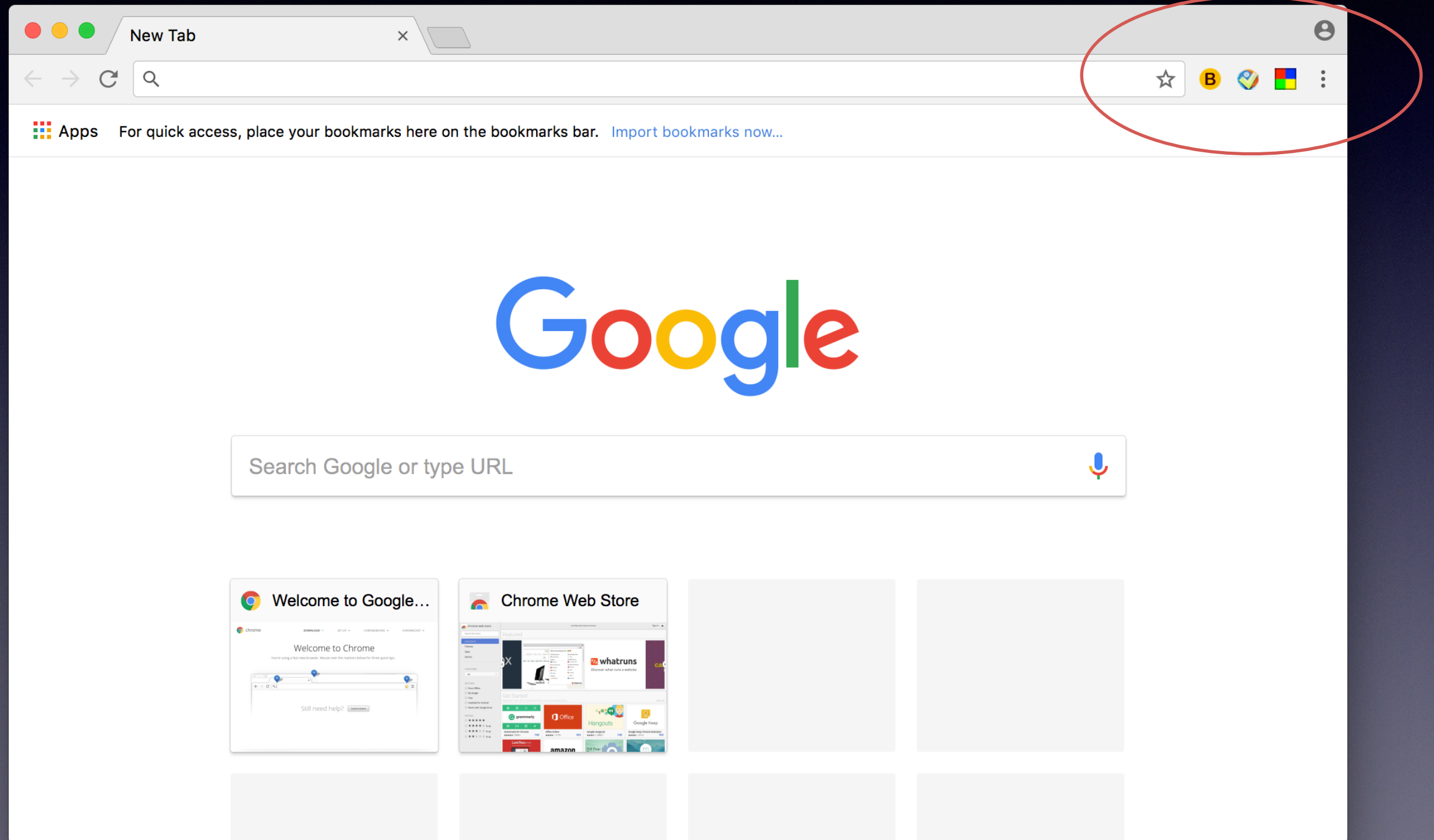
During joint forcible-entry operations, the MI and signal companies deploy in small elements to embed with maneuver units or with the BCT headquarters to perform multiple warfighting functions. Previously, the MI and signal companies had separate command posts (CPs). During the training exercise in July 2015, the MI and signal companies merged CPs, resulting in several positive outcomes. The combined CP eased the constraint of the limited personnel available to execute battle-tracking tasks. Also, colocating similar communication capabilities release and frequency modulation

Overview

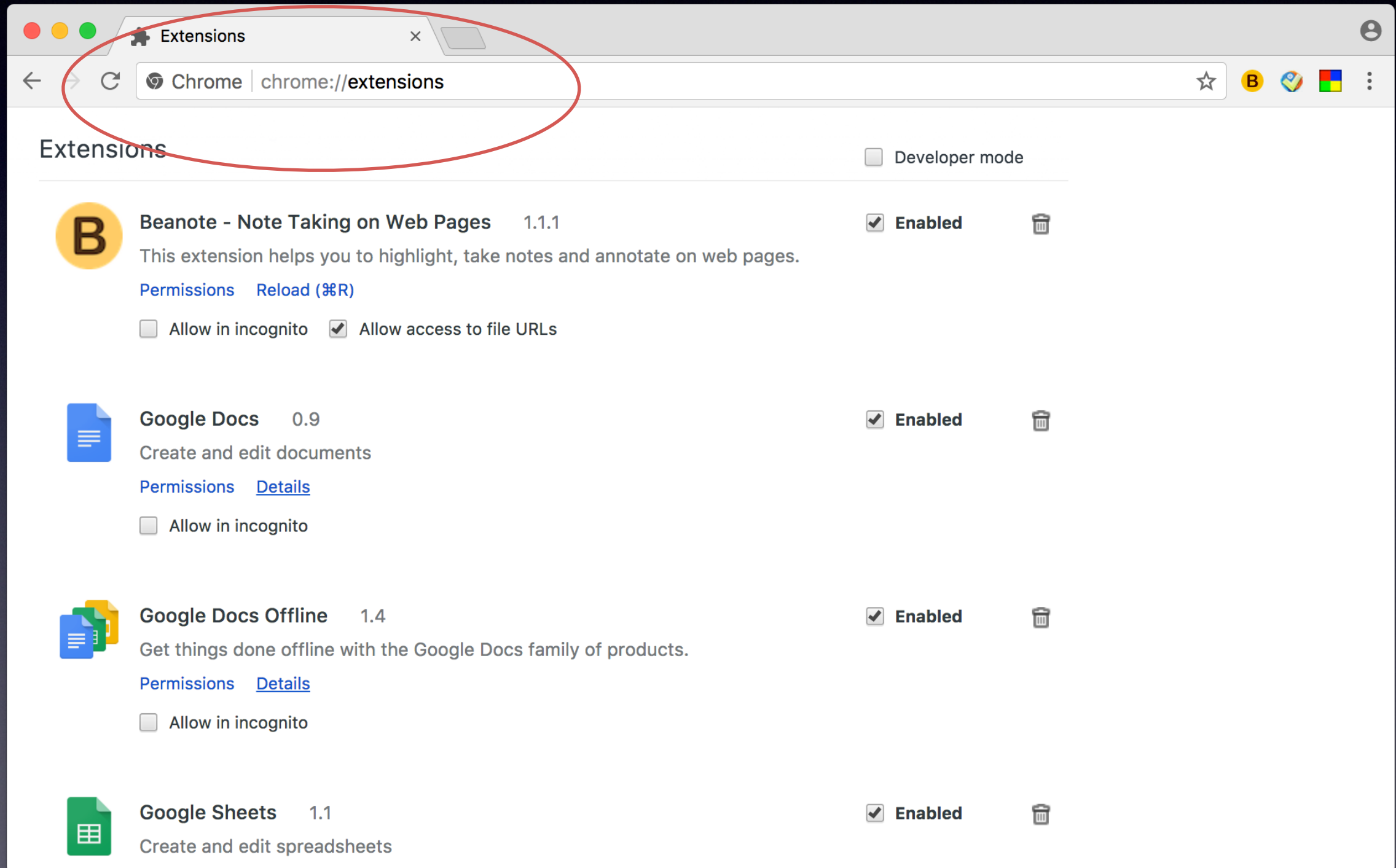
What are extensions?

Extensions are small software programs that can modify and enhance the functionality of the Chrome browser.

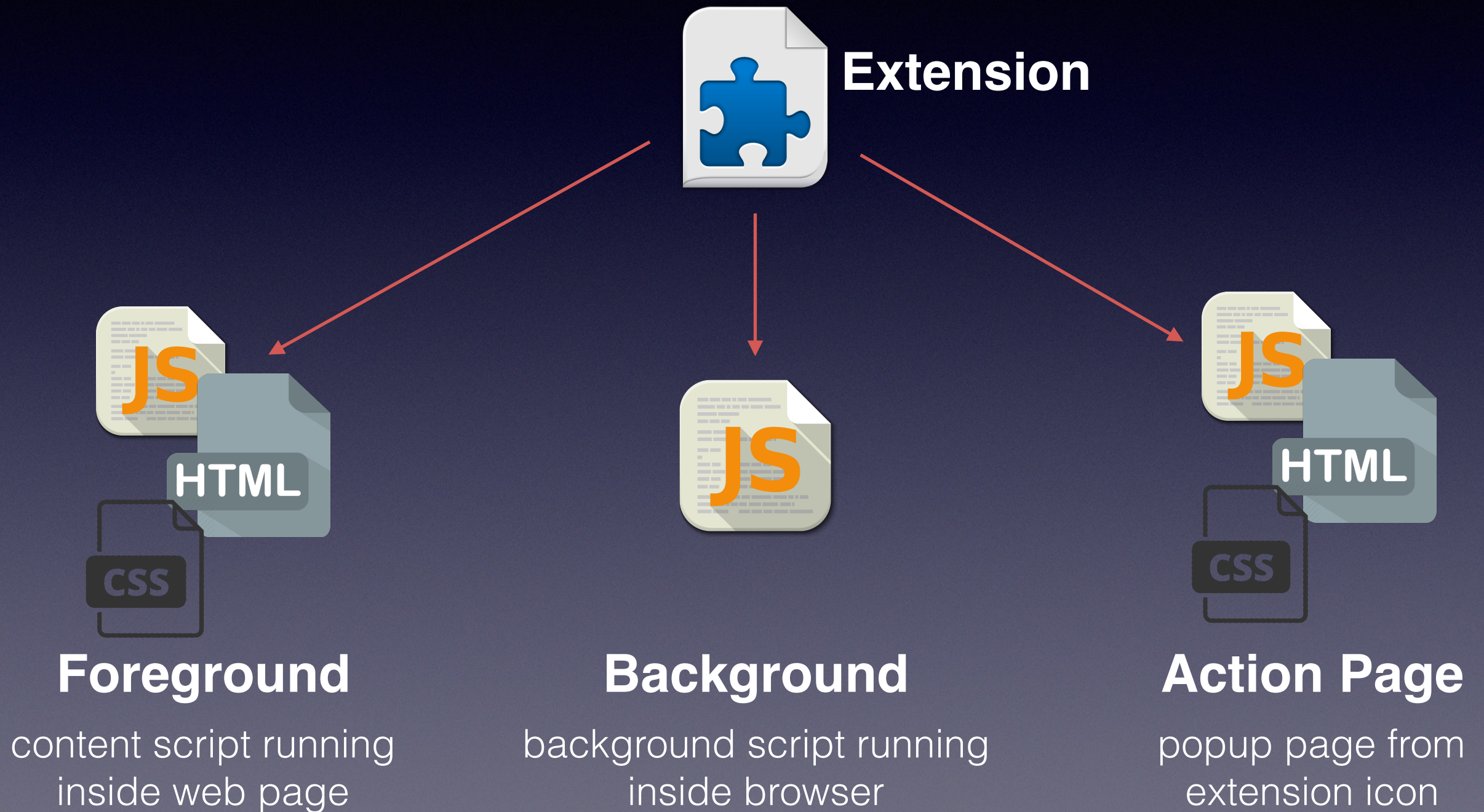
Overview



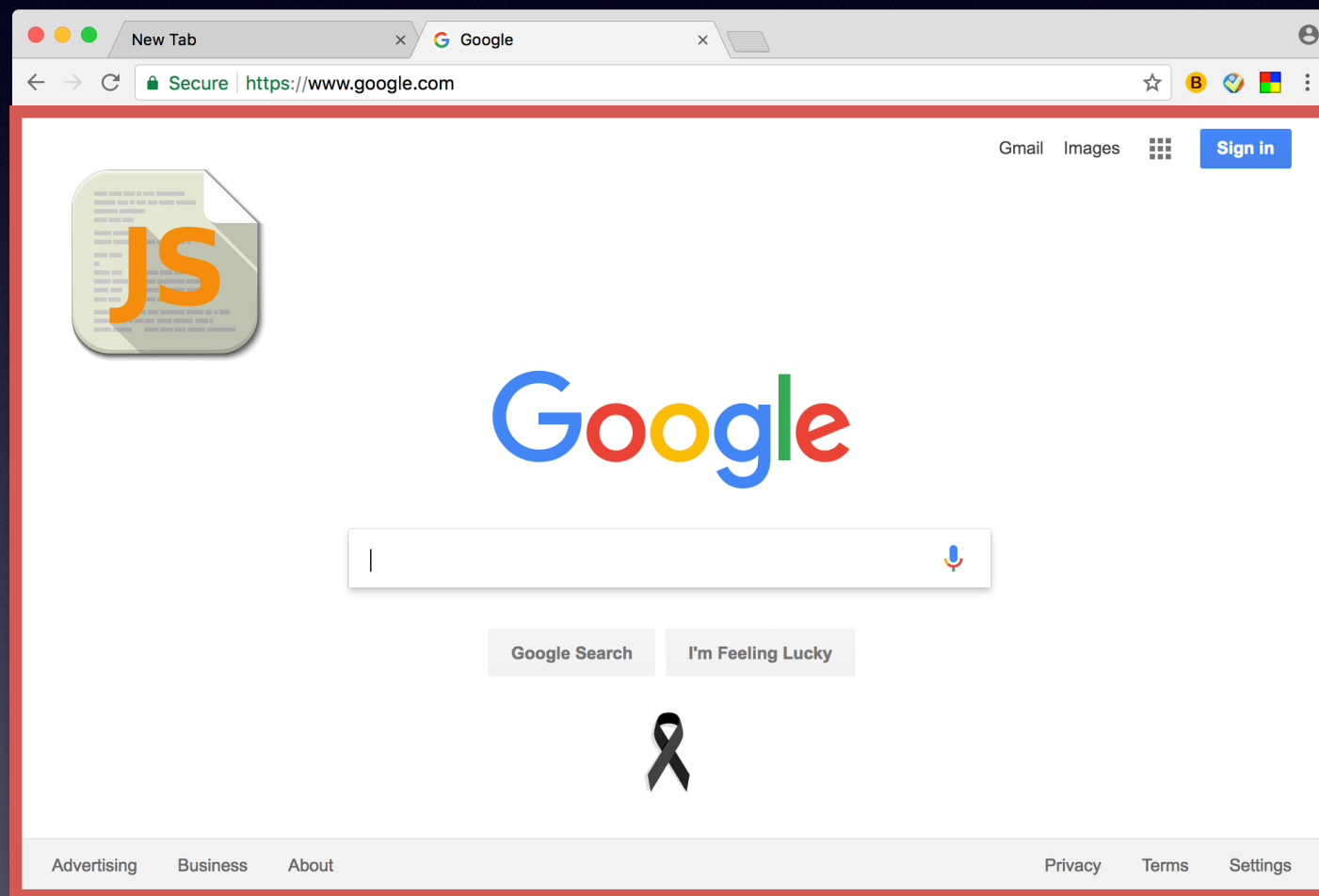
Overview



Components



Components

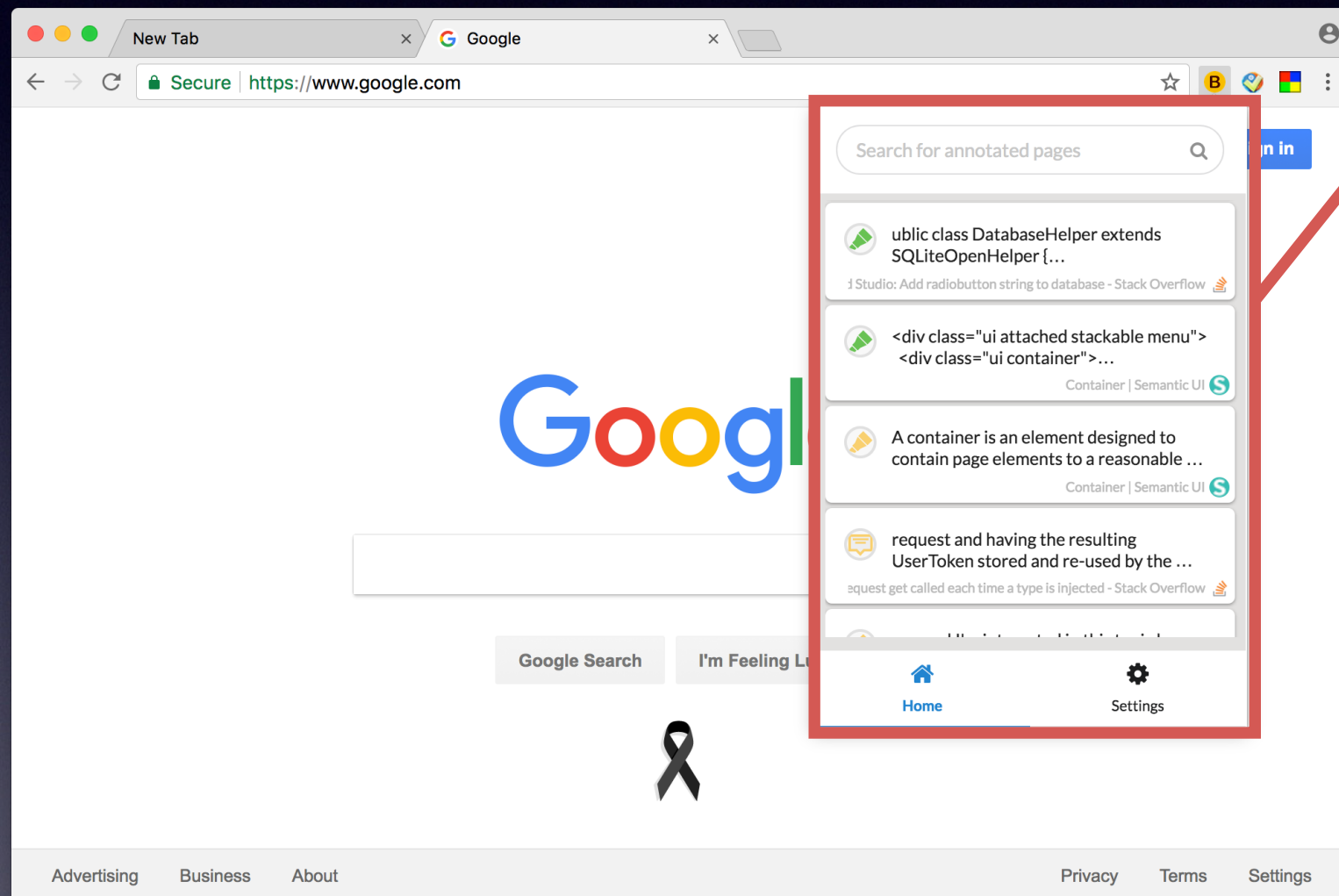


Foreground Script

- Running inside *each* webpage (tab)
- Start to run after the webpage is loaded
- Able to change any DOM element in the webpage

Just like the webpage's own javascript files

Components



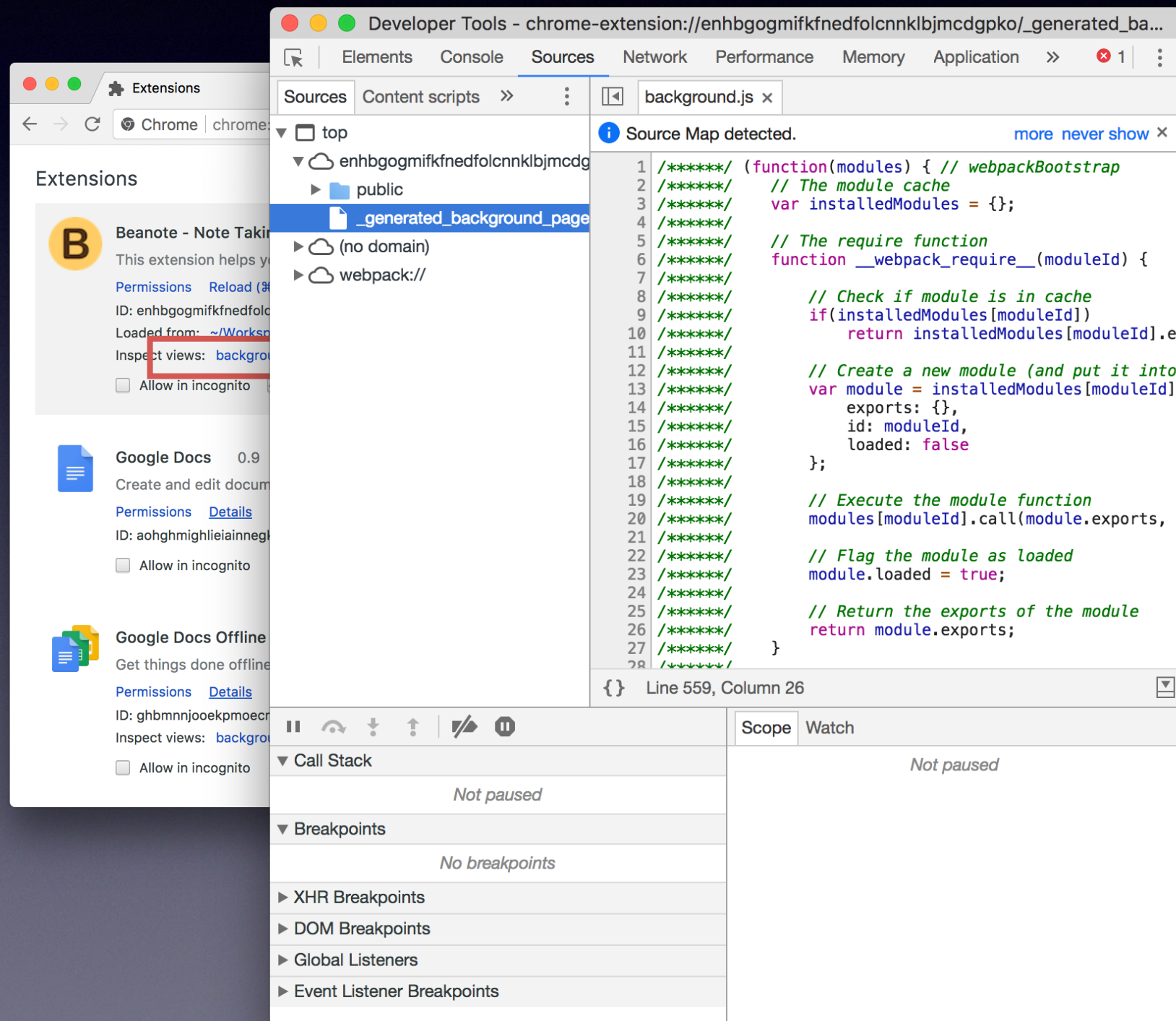
Action Page

- Popped from the extension icon
- A stand-alone webpage
- Per browser

Extension setting.

A web interface to interact with extensions

Components



Background Script

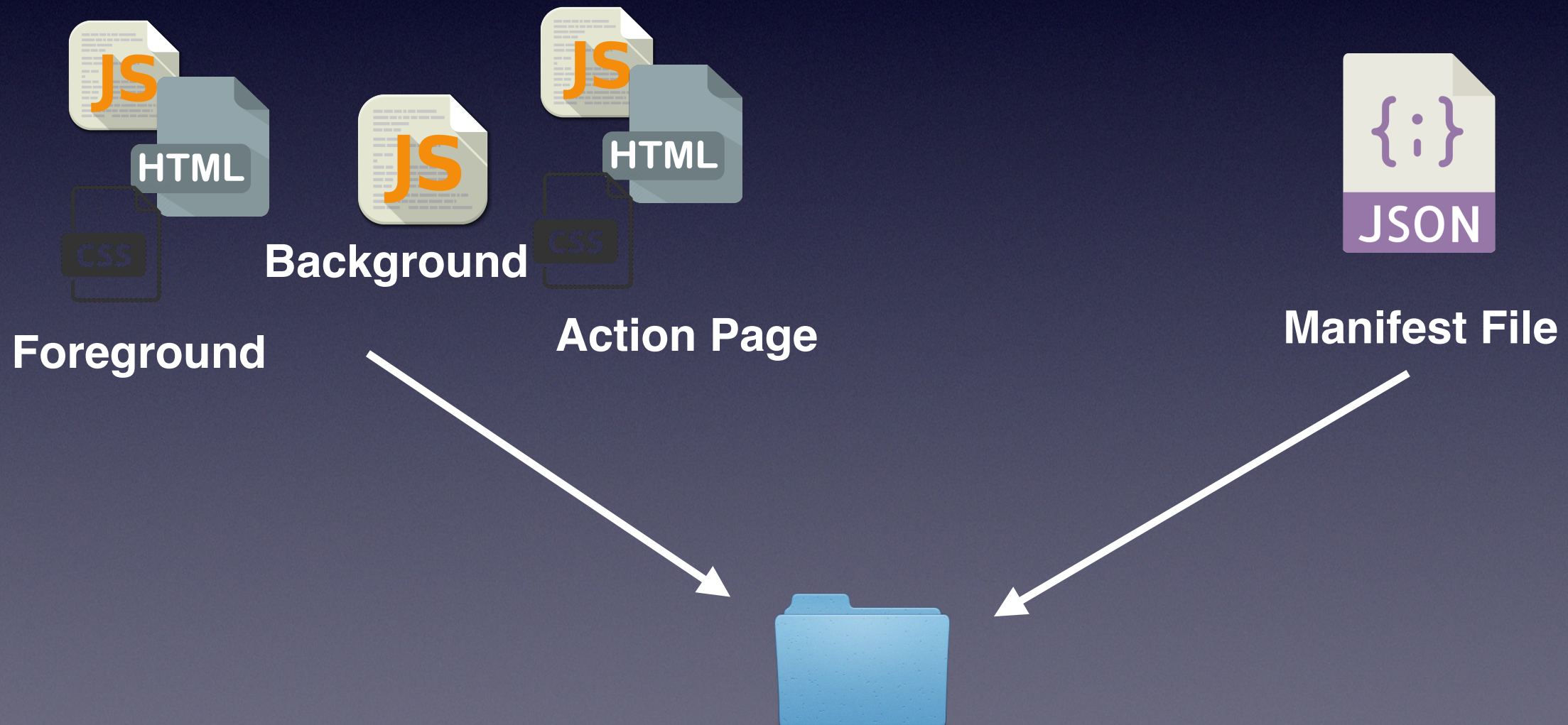
- Running in background per browser
- Access local files, storage system, network transaction etc.

Writing Chrome extension is very much like developing a web page itself.

Components

Here is no structure requirement for these files.

How to distinguish them.




```
{  
  "name": "Beanote – Note Taking on Web Pages",  
  "short_name": "Beanote",  
  "version": "1.1.1",  
  "manifest_version": 2,  
  "description": "This extension helps you to highlight, take notes and annotate on web pages.",
```

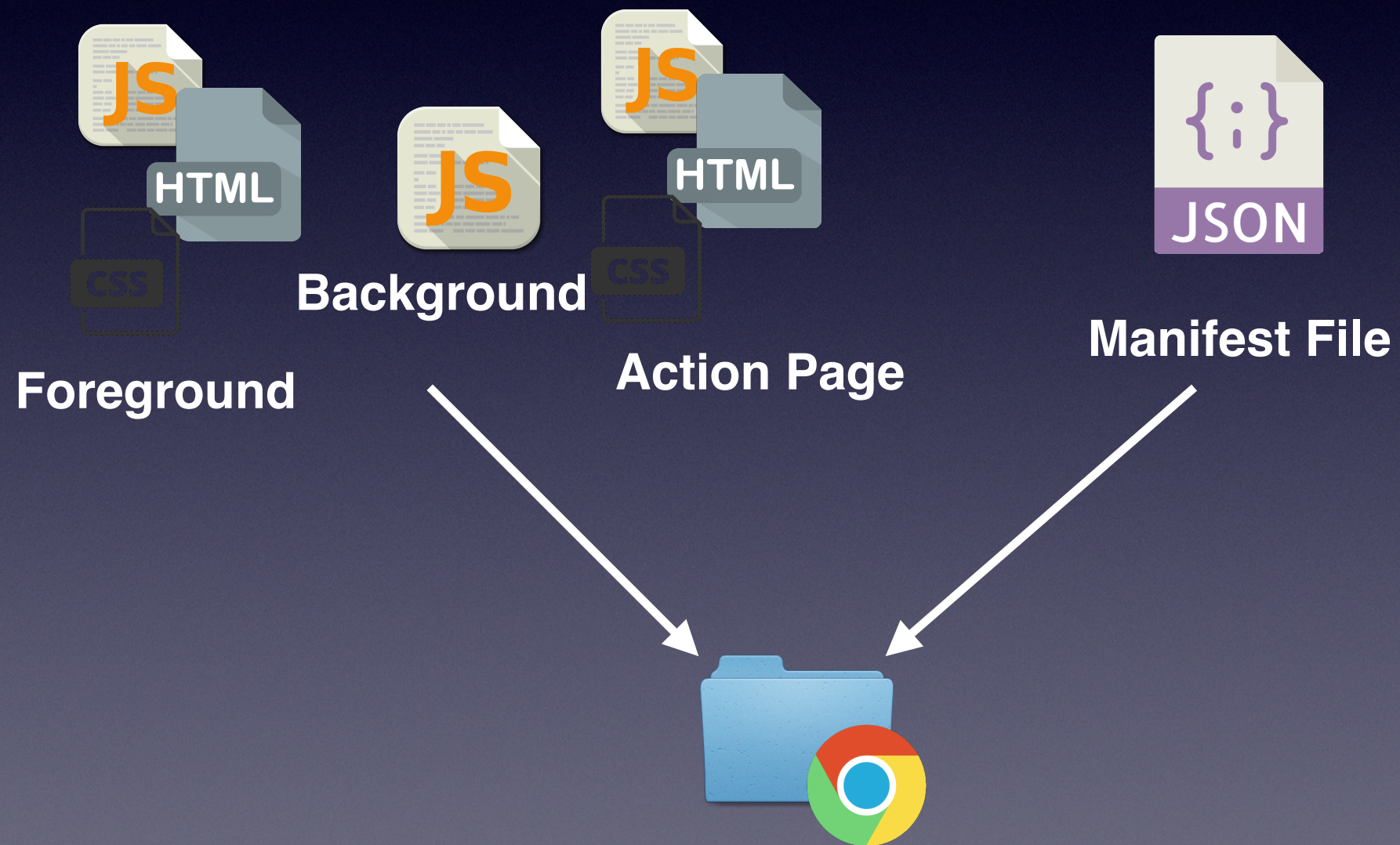
```
  "permissions": [  
    "<all_urls>",  
    "tabs",  
    "activeTab",  
    "storage",  
  ],
```

```
  "background": {  
    "scripts": [  
      "background.js"  
    ],  
    "persistent": false  
  },
```

```
  "content_scripts": [{  
    "matches": [  
      "<all_urls>"  
    ],  
    "js": [  
      "foreground.js"  
    ]  
  }],
```

```
  "browser_action": {  
    "default_title": "Beanote",  
    "default_icon": "public/icons/icon-chrome.png",  
    "default_popup": "public/actionPage/index.html"  
  }  
}
```


Components



Extensions

chrome://extensions

☆ B

Extensions

Load unpacked extension...

Pack extension...

Update extensions now

A browser action with a popup that changes the page color1.0

Change the current page color

PermissionsReload (⌘R)

ID: gfjojcglgnfjdfoeaoemgaljhjaadfc

Loaded from: ~/Downloads/set_page_color

Allow in incognito

Developer mode

Enabled

Beanote - Note Taking on Web Pages1.1.1

This extension helps you to highlight, take notes and annotate on web pages.

PermissionsReload (⌘R)

ID: enhbgogmifkfndfolcnnklbjmcdgpk

Loaded from: ~/Workspace/Beanote/beenote

Inspect views: background page (Inactive)

Allow in incognito

Allow access to file URLs

Enabled

Google Docs0.9

Create and edit documents

Enabled

Demo

- Set Page Color

[https://developer.chrome.com/extensions/examples/api/
browserAction/set_page_color.zip](https://developer.chrome.com/extensions/examples/api/browserAction/set_page_color.zip)

1. Show how to load a chrome extension
2. Show the content of each file in the folder
3. Ask about what the document here represent.

Chrome Extension



Foreground



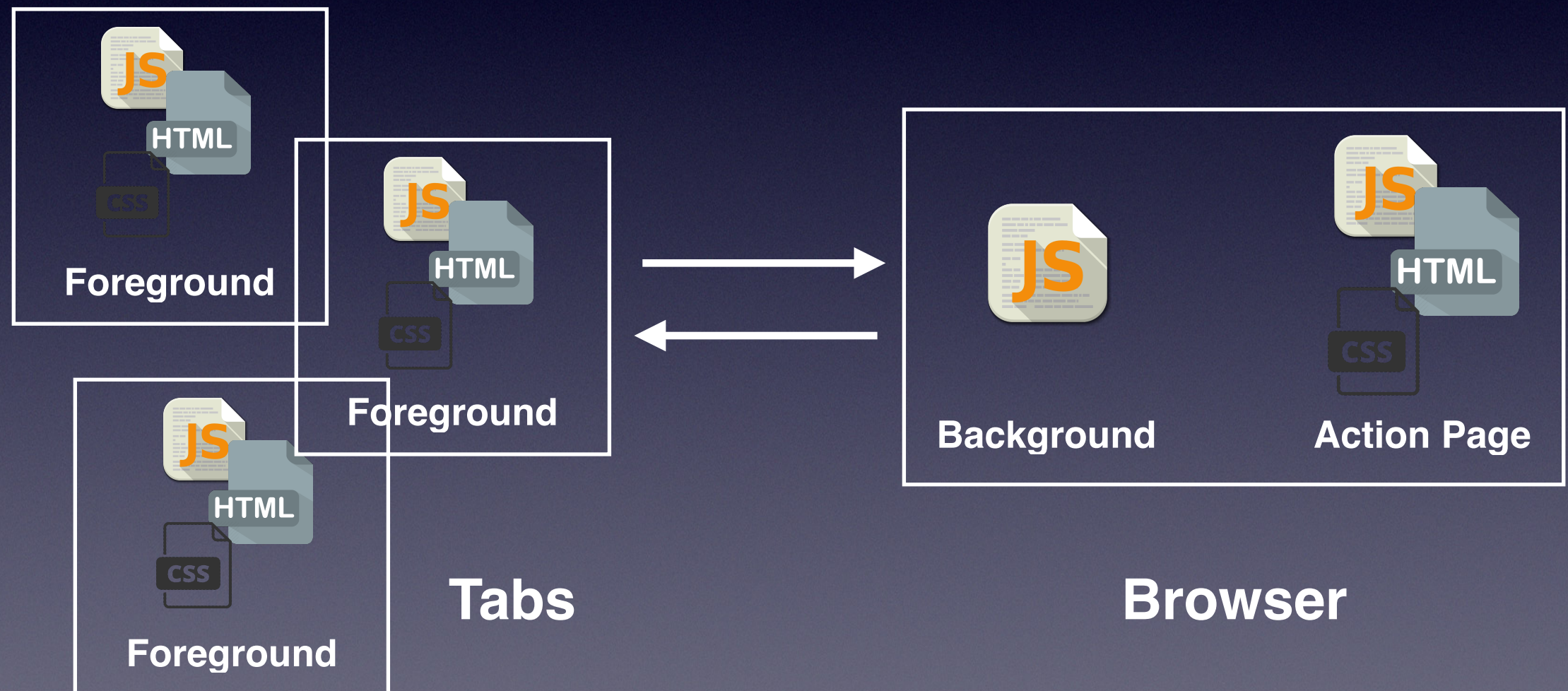
Background



Action Page

How do they communicate?

Chrome Extension: Message



Chrome Extension: Message

- From content script (foreground) to background or action page

```
chrome.runtime.sendMessage(  
    Object message,  
    Object options,  
    function responseCallback  
)
```

```
chrome.runtime.sendMessage(  
    {message: 'hello world'},  
    function(response) {  
        /* do something based on the response */  
    }  
)
```


Chrome Extension: Message

- From background or action page to content script

```
chrome.tabs.sendMessage(  
    integer tabId,  
    Object message,  
    Object options,  
    function responseCallback  
)
```

```
chrome.tabs.getCurrent(  
    function(tabInfo) {  
    }  
)
```

```
chrome.tabs.query({  
    currentWindow: true,  
    active: true  
},  
    function(tabArray) {  
    }  
)
```


Chrome Extension: Message

- Receive a message

```
chrome.runtime.onMessage.addListener(  
    function(message, sender, responseCallback) {  
    }  
)
```


Demo

- content script and message passing

Shadow DOM

- DOM encapsulation

Shadow DOM

```
div = document.createElement('div')  
shadowDom = div.createShadowRoot()
```