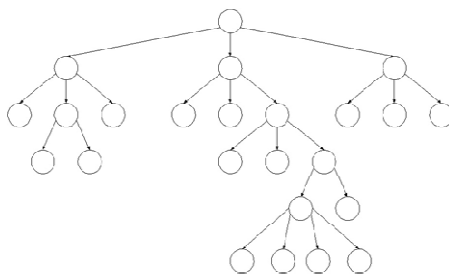


Kako da razumijete DOM

DOM čvorovi

DOM drvo je sastavljeno od čvorova ([Node](#))

Q: Da li čvorovi u DOM imaju pokazivače ka djeci kao drveća u jezicima C/C++/Java/Pasca?

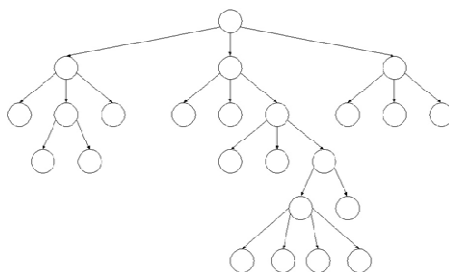


DOM čvorovi

DOM drvo je sastavljeno od čvorova ([Node](#))

Q: Da li čvorovi u DOM imaju pokazivače ka djeci kao drveća u jezicima C/C++/Java/Pascal?

A: DA!



Svojstva čvorova

Svojstvo	Opis
textContent	Tekstualni sadržaj čvora i njegovih potomaka (writeable)
childNodes	Niz djece čvora (prazan ako je element)
parentNode	Referenca na roditeljski čvor

```

<body>
  <h1>My favorites</h1>
  <section>
    <p>Strawberries</p>
    <p>Chocolate</p>
  </section>
</body>

```

Ko je roditelj ([parentNode](#)) elementa [<section>](#)?

parentNode

```
> section = document.querySelector('section');
< ▶ <section>...</section>
> section.parentNode
< ▶ <body>...</body>
```

```
<body>
  <h1>My favorites</h1>
  <section>
    <p>Strawberries</p>
    <p>Chocolate</p>
  </section>
```

parentNode elementa
<section> je <body>.

Ko su childNodes elementa
<section>?

childNodes

```
> section = document.querySelector('section');
< ▶ <section>...</section>
> section.childNodes
< ▶ [text, p, text, p, text]
> section.childNodes.length
< 5
```

???

```
<body>
  <h1>My favorites</h1>
  <section>
    <p>Strawberries</p>
    <p>Chocolate</p>
  </section>
```

Zašto section ima
petoro djece, a ne
samo dvoje?

TextNode

Pored čvorova tipa [Element](#), DOM takođe sadrži čvorve tipa [Text](#). Tekst u HTML-u, **uključujući bjeline**, se sadrži u tekst-čvorvima:

```
<body>
  <h1>My favorites</h1>
  <section>

    <p>Strawberries</p>

    <p>Chocolate</p>
  </section>
</body>
```

TextNode

Sav tekst u HTML-u, **uključujući bjeline**, se sadrži u tekst-čvorovima ([Text](#) node):

```
<body>
  <h1>My favorites</h1>
  <section>
    <p>Strawberries</p>
    <p>Chocolate</p>
  </section>
</body>
```

DOM i Text nodes

DOM se sastoji od čvorova tipa [Node](#); postoji više podtipova [Node](#).

- [Element](#): HTML (ili SVG) elementi DOM-a
- [Text](#): Tekstualni dio u DOM, uključuje i bjeline
 - [Text](#) čvorovi ne mogu imati djecu (uvijek su listovi)
- [Comment](#): HTML komentari
- ([more](#))

Tip čvora se čuva u svojstvu [nodeType](#)

Obilazak DOM

Q: Kako da šampamo sve čvorove u DOM?

Obilazak DOM

Q: Kako da šampamo sve čvorove u DOM?

A: Rekurzivni obilazak DOM drveta:

```
function walkTree(root, level) {  
  if (root.nodeType === Node.TEXT_NODE) {  
    console.log(level + 'text:' + root.textContent);  
  } else {  
    console.log(level + root.nodeName);  
  }  
  for (const child of root.childNodes) {  
    walkTree(child, level + "  ");  
  }  
}  
walkTree(document.querySelector('html'), "");
```

What's the point?

- Ako imamo document.querySelector koji nama dopušta da uzmemo sve DOM elemente...
- I ako možemo promijeniti HTML po potrebi dodavanjem classes/ids/elements/itd ...

Q: Kada ćemo uopšte poželjeti da obiđemo čitav DOM?

What's the point?

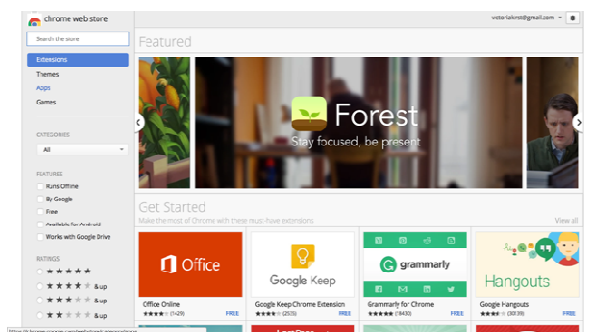
- Ako imamo document.querySelector koji nama dopušta da uzmemo sve DOM elemente...
- I ako možemo promijeniti HTML po potrebi dodavanjem classes/ids/elements/itd ...

Q: Kada ćemo uopšte poželjeti da obiđemo čitav DOM?

**A: Ili u ekstenzijama browsera ili u Web Console
(tj. manipulišemo sa tuđim stranicama)**

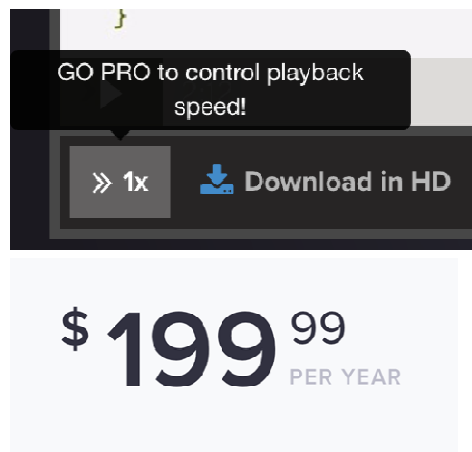
Browser extensions

- Dodatak koji proširuje funkcionalnost browsera
- Dio koda napisan u JavaScript-u koji se umeće (injected) u veb-stranicu prije ili poslije njenog učitavanja



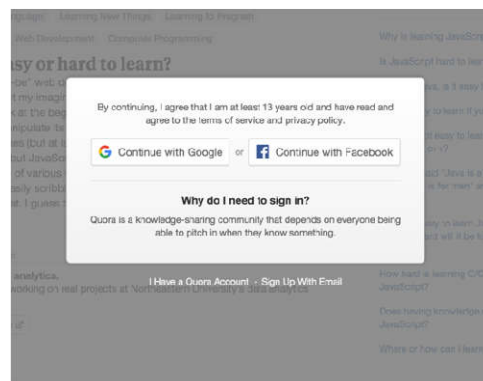
Primjer: Egghead

[Dan Abromov's Redux videos](#)



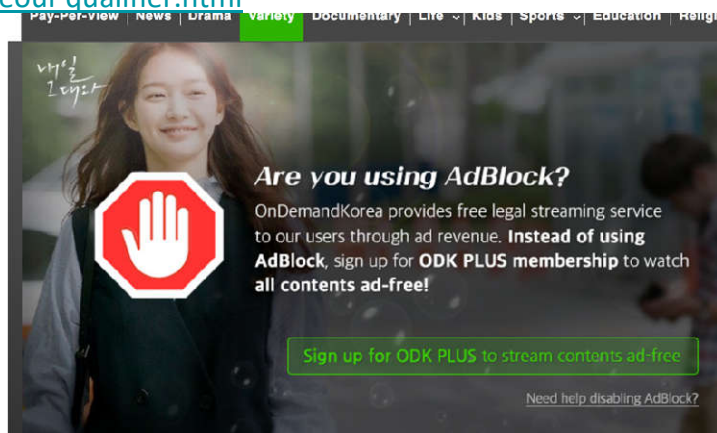
Primjer: Quora signin wall

<https://www.quora.com/Why-is-learning-JavaScript-so-hard>



Primjer: Adblock block

<http://www.ondemandkorea.com/kpop-star-season-6-seoul-qualifier.html>



Content scripts

- Tip ekstenzije koji se izvršava u kontekstu browsera – može pristupiti DOM-u stranice
 - Koncept je isti za [Chrome](#), [Firefox](#), vjerovatno i za ostale browsera

Najčešće se sastoji od:

- manifest.json: sadrži title, settings itd. za skript
- page.js: extension file

Primjer: manifest.json

```
{
  "name": "Empty Chrome Extension",
  "version": "1.0",
  "description": "",
  "content_scripts": [
    {
      "matches": ["<all_urls>"],
      "js": ["page.js"]
    }
  ],
  "icons": {
    "16": "pizza.png",
    "48": "pizza.png",
    "128": "pizza.png"
  },
  "manifest_version": 2
}
```

Primjer: page.js

```
page.js
1 console.log('Empty Extension test message');
2
```