Extracting Data from the Web
Part 2: **Web Scraping**

Garrett Grolemund
Data Scientist, Educator
November 2016

1. **APIs** (www.rstudio.com/resources/webinars/)
2. **Web Scraping** (Today)
    a. Basic Idea - HTML, CSS
    b. rvest
    c. SelectorGadget

rvest SelectorGadget Vignette

400 words, <5 minutes to read

cran.r-project.org/web/packages/rvest/vignettes/selectorgadget.html

Extracting Data from the Web

Scott Chamberlain, Karthik Ram, Me

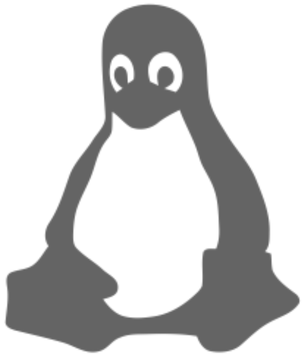
github.com/ropensci/user2016-tutorial

Web Scraping

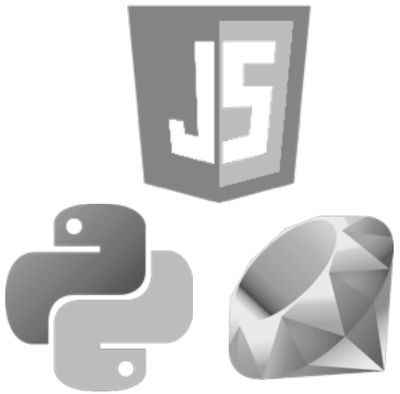
APIs

An interface for your code to interact with a piece of software:

Database



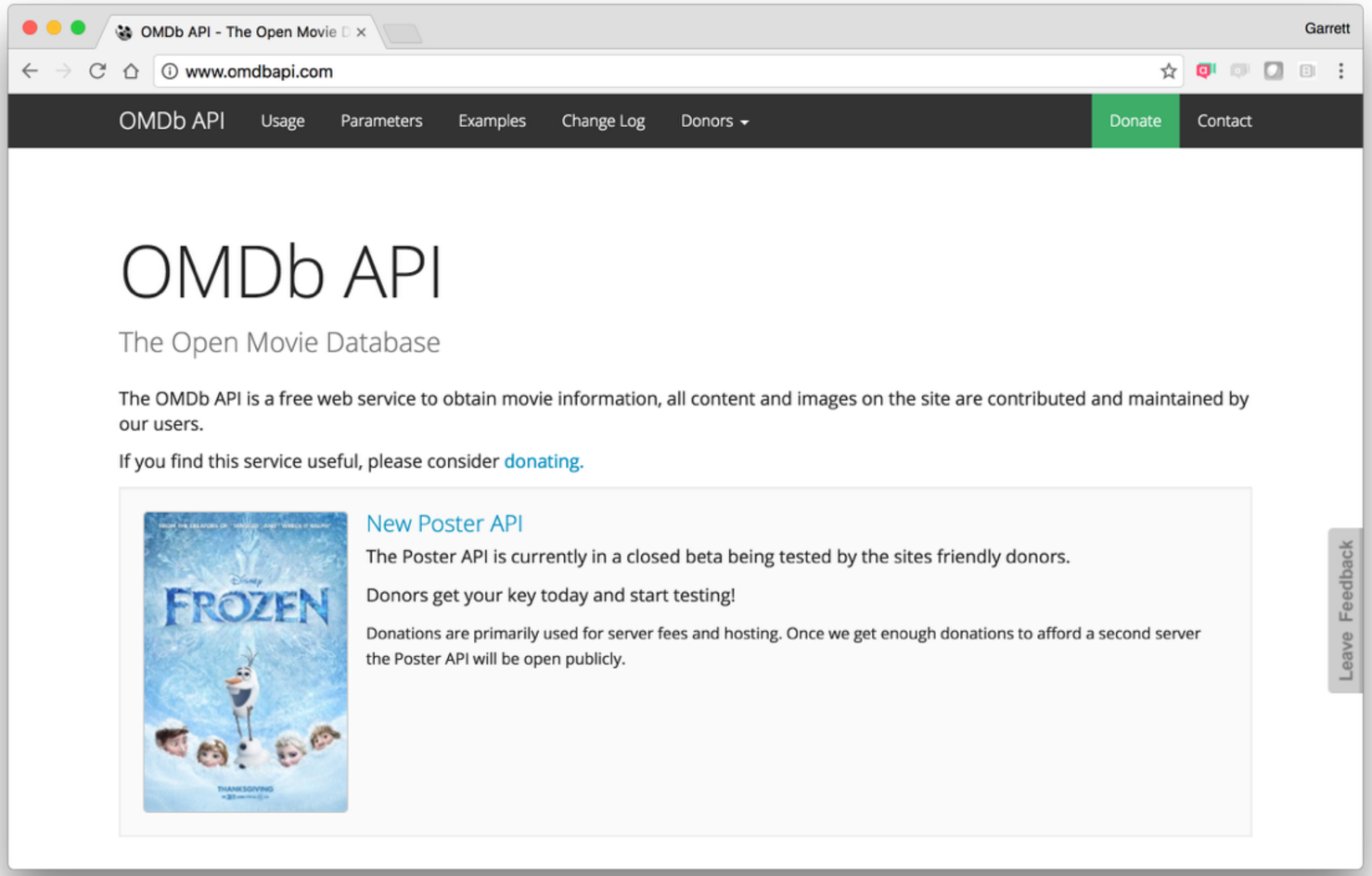
OS



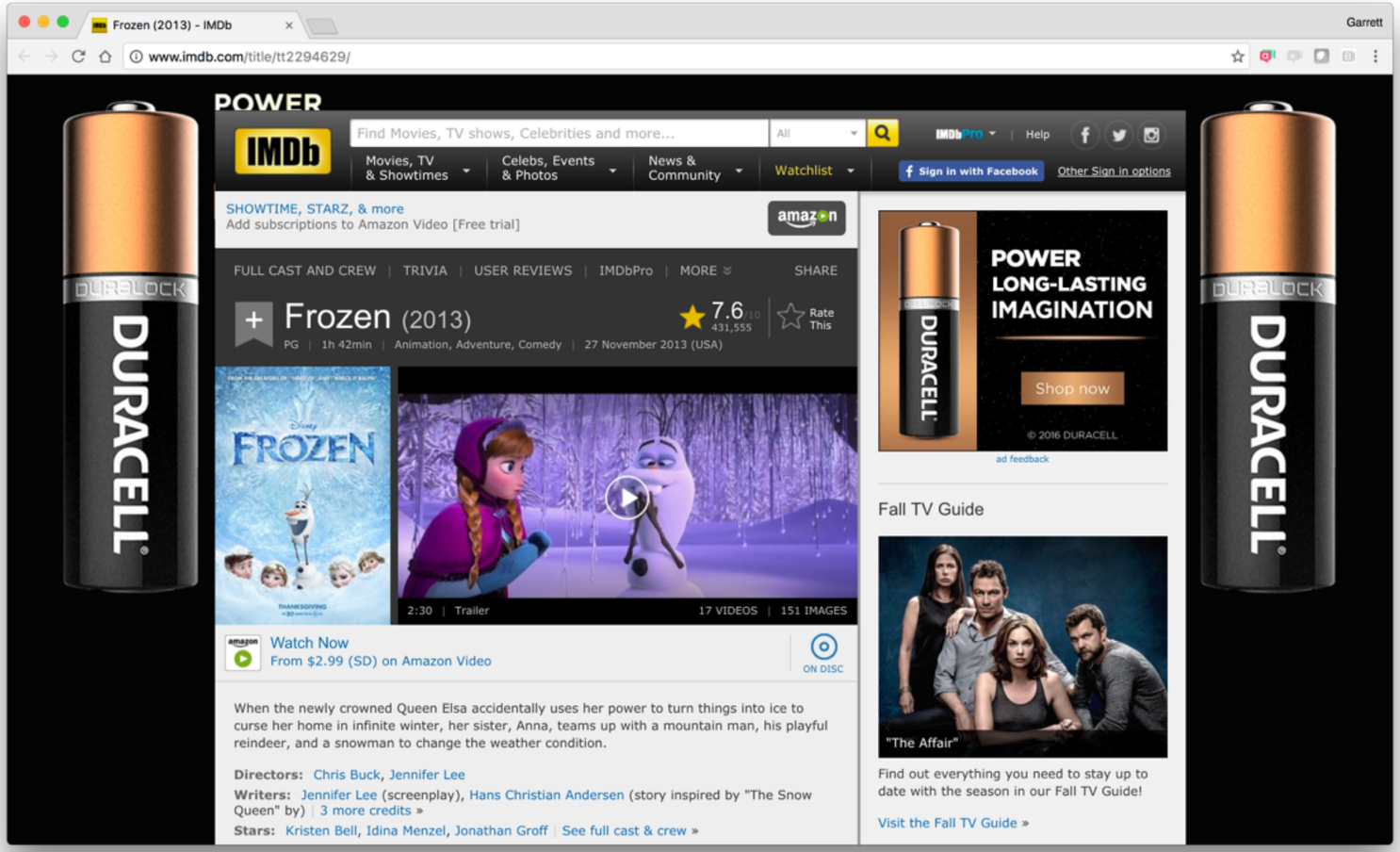
Software package

Web Application

www.omdbapi.com



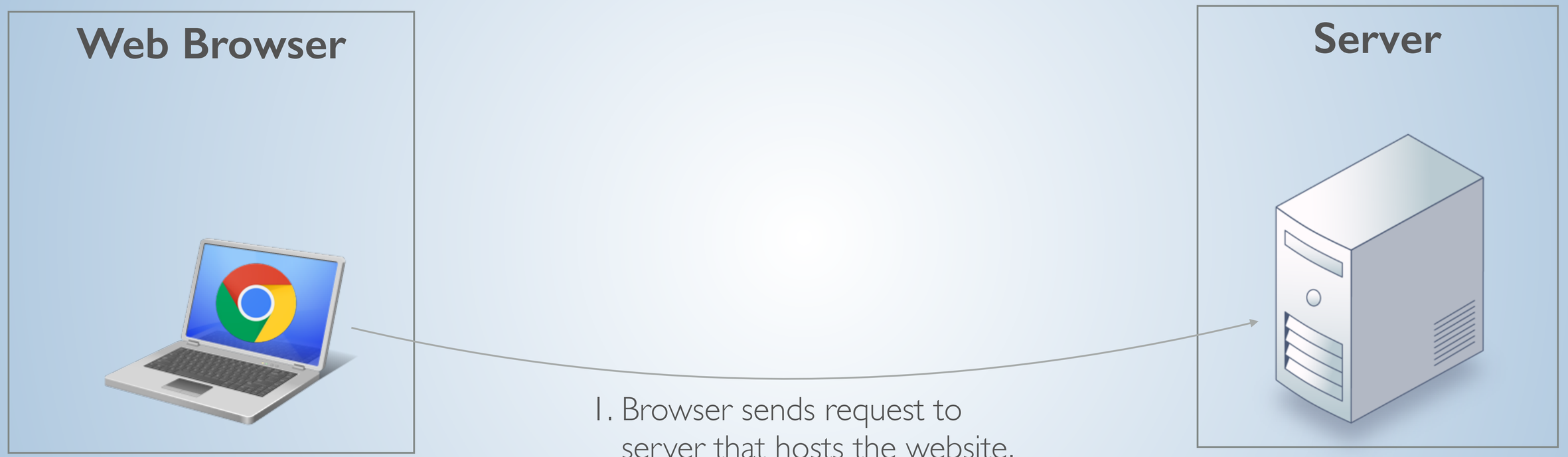
www.imdb.com



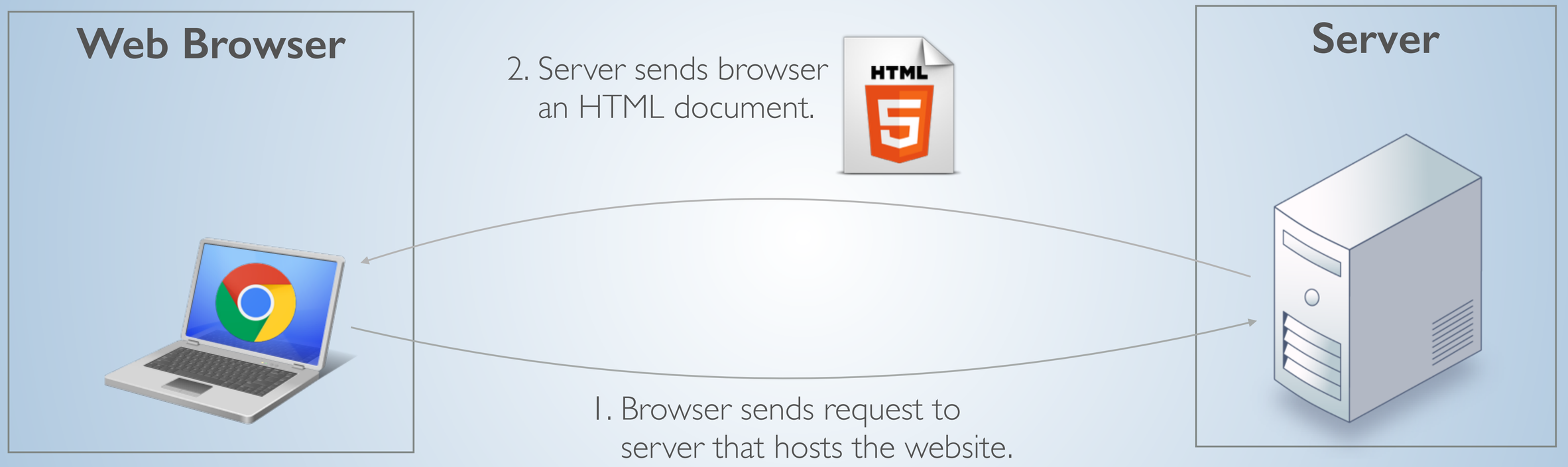
No API

Web

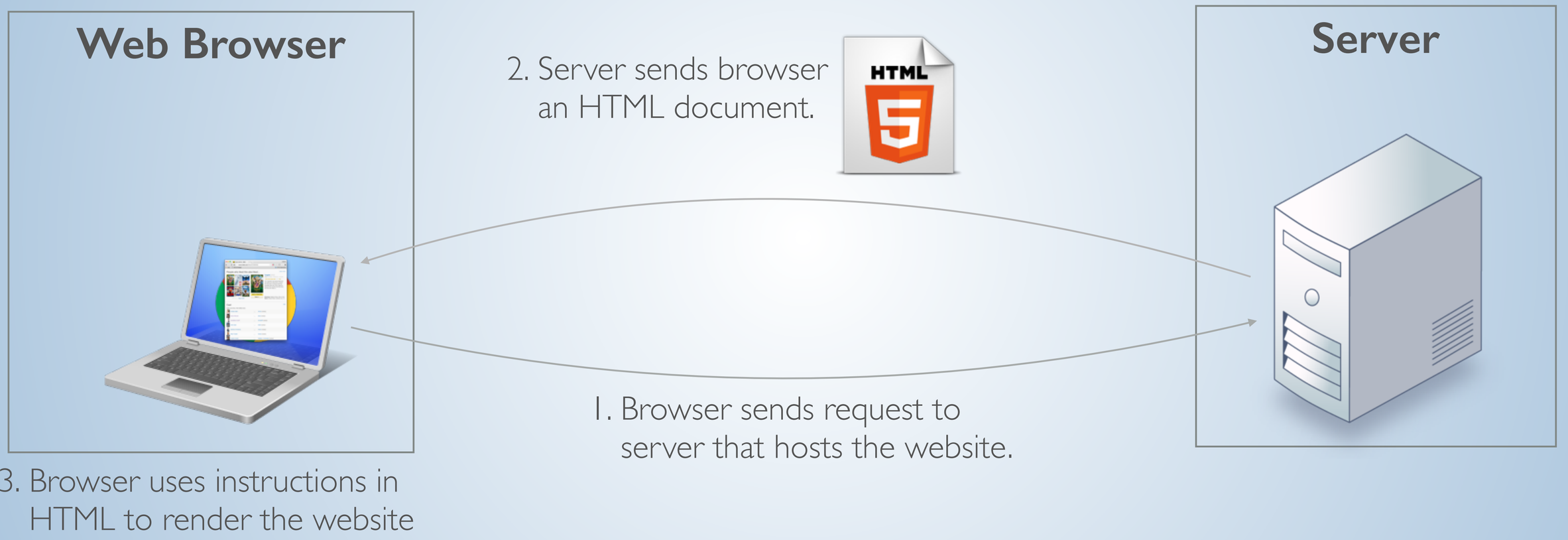
HTML (Review)


Web Browser
Server
1. Browser sends request to server that hosts the website.

HTML (Review)


Web Browser
2. Server sends browser an HTML document.
HTML
Server
1. Browser sends request to server that hosts the website.

HTML (Review)


Web Browser
2. Server sends browser an HTML document.
HTML
Server
1. Browser sends request to server that hosts the website.
3. Browser uses instructions in HTML to render the website

HTML (Review)

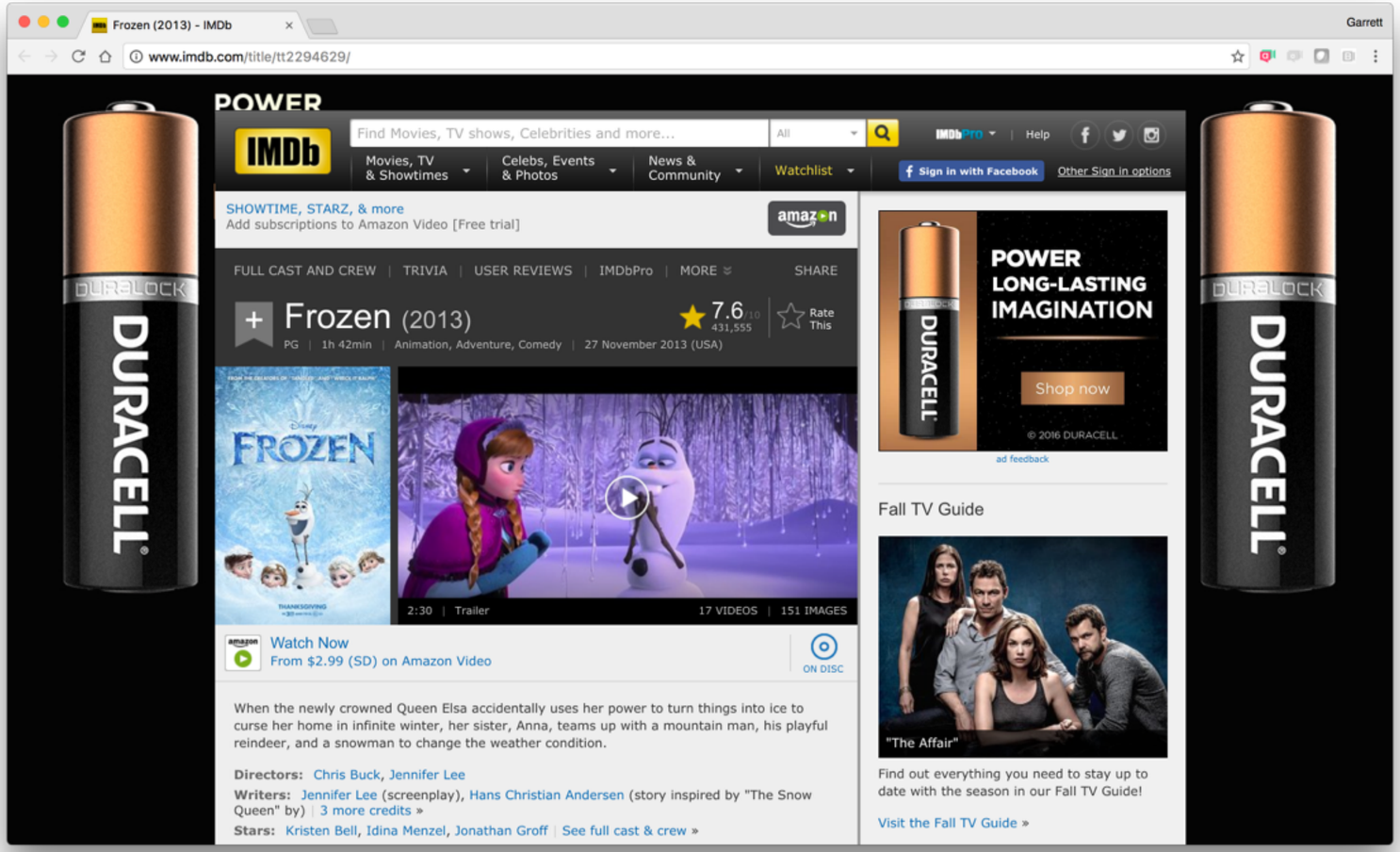

Web Browser
Server
HTML
<html>
<head>
<title>Title</title>
<link rel="icon" type="icon" href="http://a" />
<link rel="icon" type="icon" href="http://b" />
<script type="text/javascript">
var ue_t0=window.ue_t0||+new Date();
</script>
</head>
<body>

Click here now.
Frozen

<table style="width:100%">
<tr>
<td>Kristen</td>
<td>Bell</td>
</tr>
<tr>
<td>Idina</td>
<td>Menzel</td>
</tr>
</table>

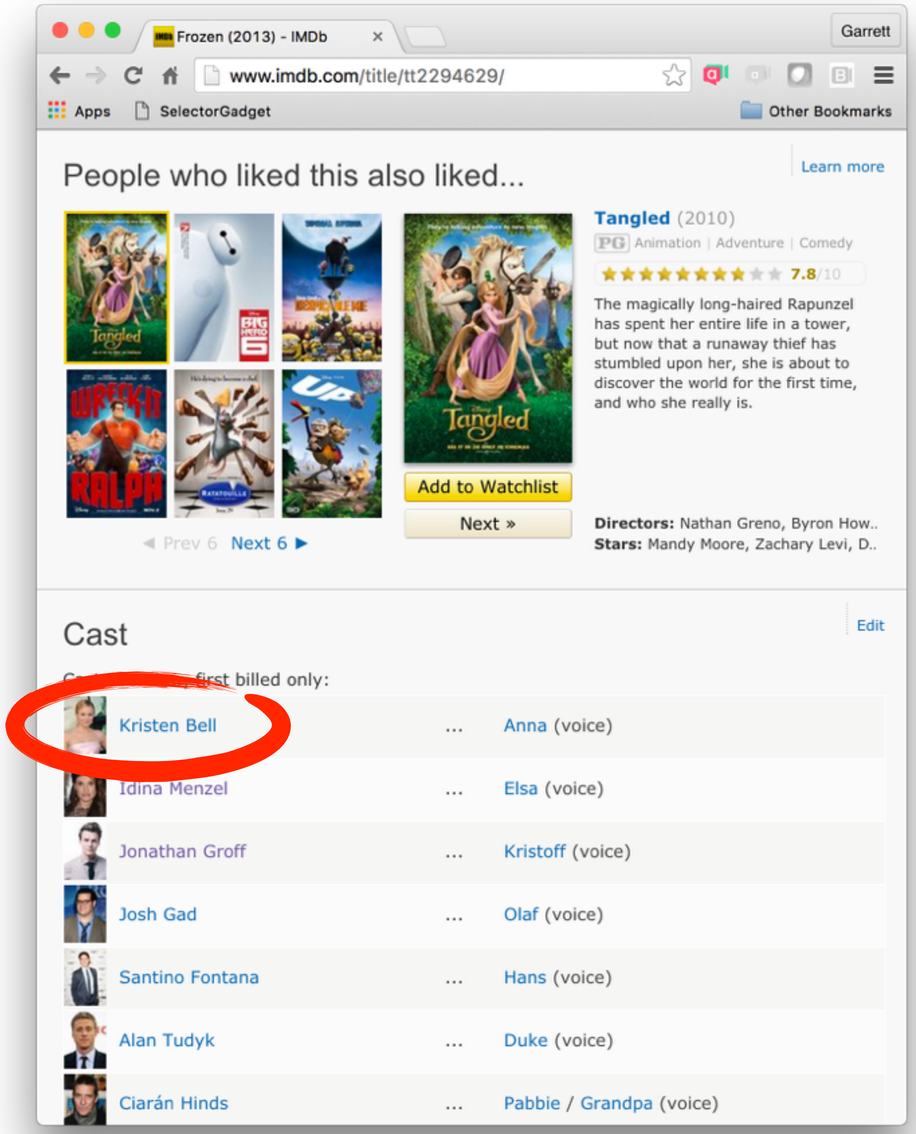
</body>
</html>
quest to
the website.
3. Browser uses instructions
HTML to render the webs

www.imdb.com



No API

Web


Frozen (2013) - IMDb
Garrett
www.imdb.com/title/tt2294629/
Apps
SelectorGadget
Other Bookmarks
People who liked this also liked...
Learn more
Tangled (2010)
PG Animation | Adventure | Comedy
7.8/10
The magically long-haired Rapunzel has spent her entire life in a tower, but now that a runaway thief has stumbled upon her, she is about to discover the world for the first time, and who she really is.
Add to Watchlist
Next »
Directors: Nathan Greno, Byron How..
Stars: Mandy Moore, Zachary Levi, D..
◄ Prev 6 Next 6 ►
Cast
Edit
first billed only:
Kristen Bell ... Anna (voice)
Idina Menzel ... Elsa (voice)
Jonathan Groff ... Kristoff (voice)
Josh Gad ... Olaf (voice)
Santino Fontana ... Hans (voice)
Alan Tudyk ... Duke (voice)
Ciarán Hinds ... Pabbie / Grandpa (voice)


Frozen (2013) - IMDb
view-source:www.imdb.com
Garrett
view-source:www.imdb.com/title/tt2294629/
Apps
SelectorGadget
Other Bookmarks
3933
3934 <table class="cast_list">
3935 <tr><td colspan="4" class="castlist_label">Cast overview, first billed only:</td></tr>
3936 <tr class="odd">
3937 <td class="primary_photo">
3938 <a href="/name/nm0068338/?ref_=tt_cl_i1"
3939 >
</td>
3940 <td class="itemprop" itemprop="actor" itemscope itemtype="http://schema.org/Person">
3941 <a href="/name/nm0068338/?ref_=tt_cl_t1"
3942 itemprop='url'>Kristen Bell
3943
</td>
3944 <td class="ellipsis">
3945 ...
3946 </td>
3947 <td class="character">
3948
3949 Anna
3950
3951
3952 (voice)
3953


Web

Strategy

1. Extract information from the HTML DOM
2. Use the structure of HTML to find the information

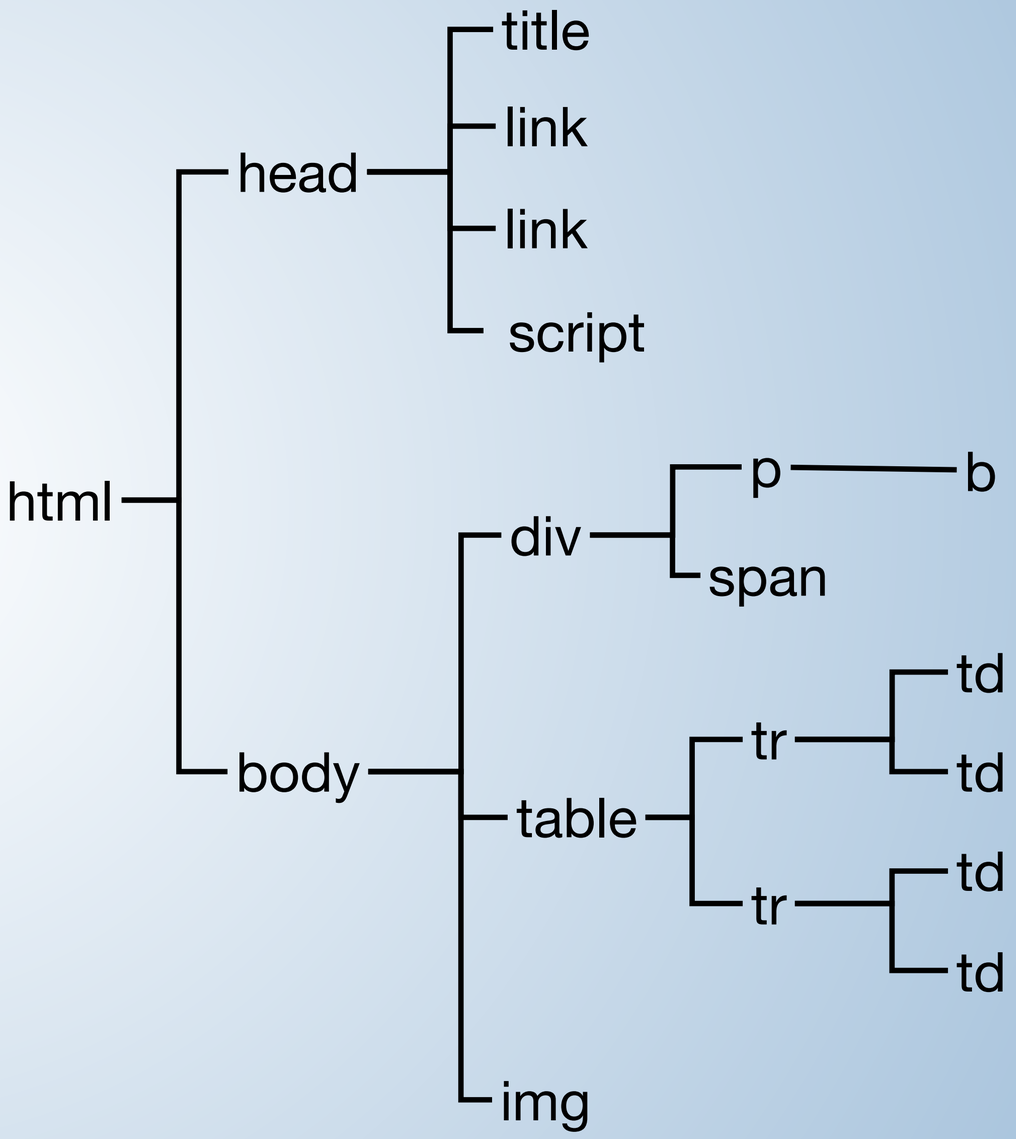
HTML (Review)

Each element in the page is created by a tag.

```
<a href="http://github.com">GitHub</a>
```

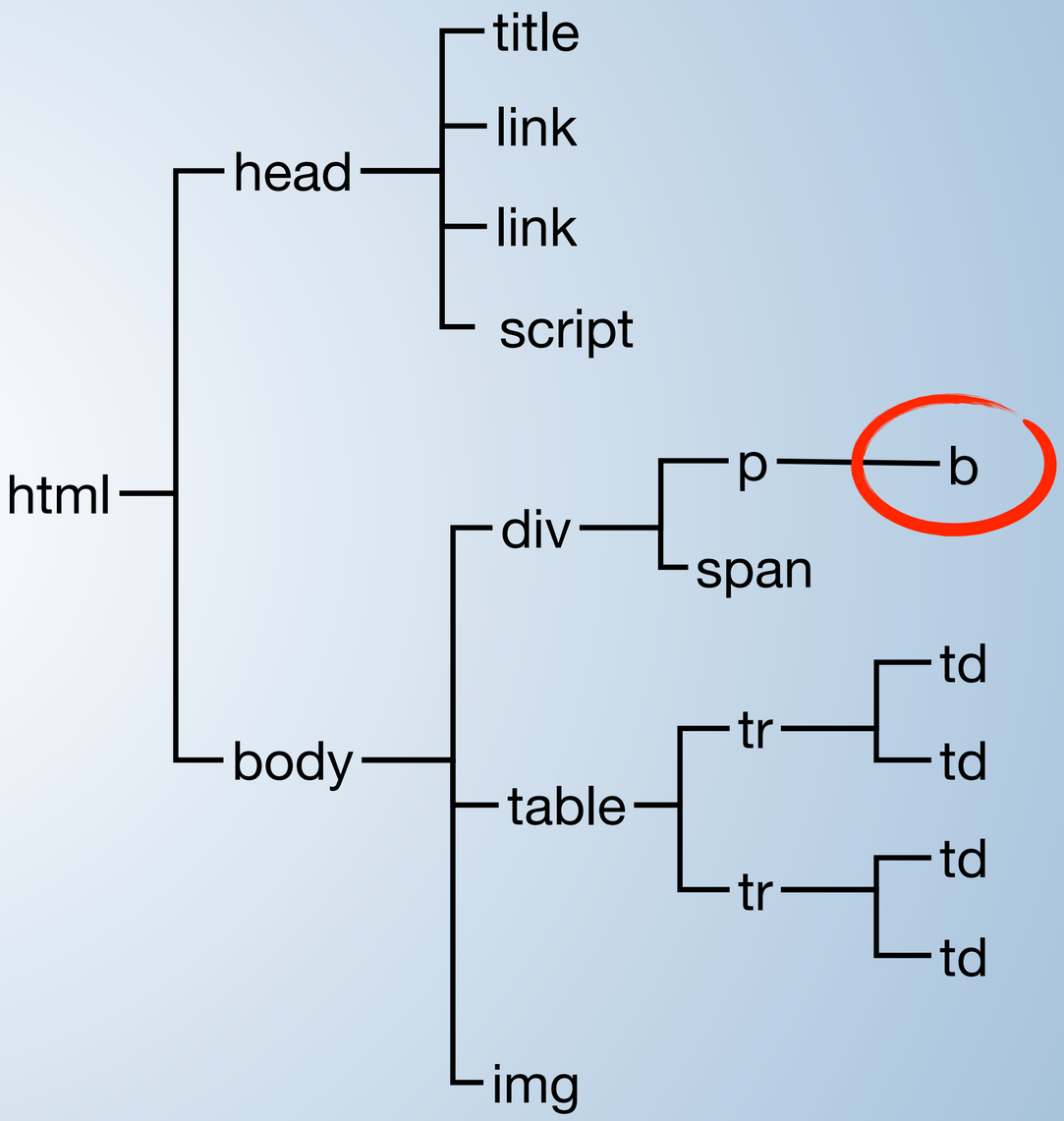
HTML (Review)


```
<html>
  <head>
    <title>Title</title>
    <link rel="icon" type="icon" href="http://a" />
    <link rel="icon" type="icon" href="http://b" />
    <script type="text/javascript">
      var ue_t0=window.ue_t0||+new Date();
    </script>
  </head>
  <body>
    <div>
      <p>Click <b>here</b> now.</p>
      <span>Frozen</span>
    </div>
    <table style="width:100%">
      <tr>
        <td>Kristen</td>
        <td>Bell</td>
      </tr>
      <tr>
        <td>Idina</td>
        <td>Menzel</td>
      </tr>
    </table>
    <img src="http://ia.media-imdb.com/images.png"/>
  </body>
</html>
```



HTML (Review)


```
<html>
  <head>
    <title>Title</title>
    <link rel="icon" type="icon" href="http://a" />
    <link rel="icon" type="icon" href="http://b" />
    <script type="text/javascript">
      var ue_t0=window.ue_t0||+new Date();
    </script>
  </head>
  <body>
    <div>
      <p>Click <b>here</b> now.</p>
      <span>Frozen</span>
    </div>
    <table style="width:100%">
      <tr>
        <td>Kristen</td>
        <td>Bell</td>
      </tr>
      <tr>
        <td>Idina</td>
        <td>Menzel</td>
      </tr>
    </table>
    <img src="http://ia.media-imdb.com/images.png"/>
  </body>
</html>
```



Which HTML tag surrounds Kristin Bell’s name?

Which HTML tag surrounds Kristin Bell's name?

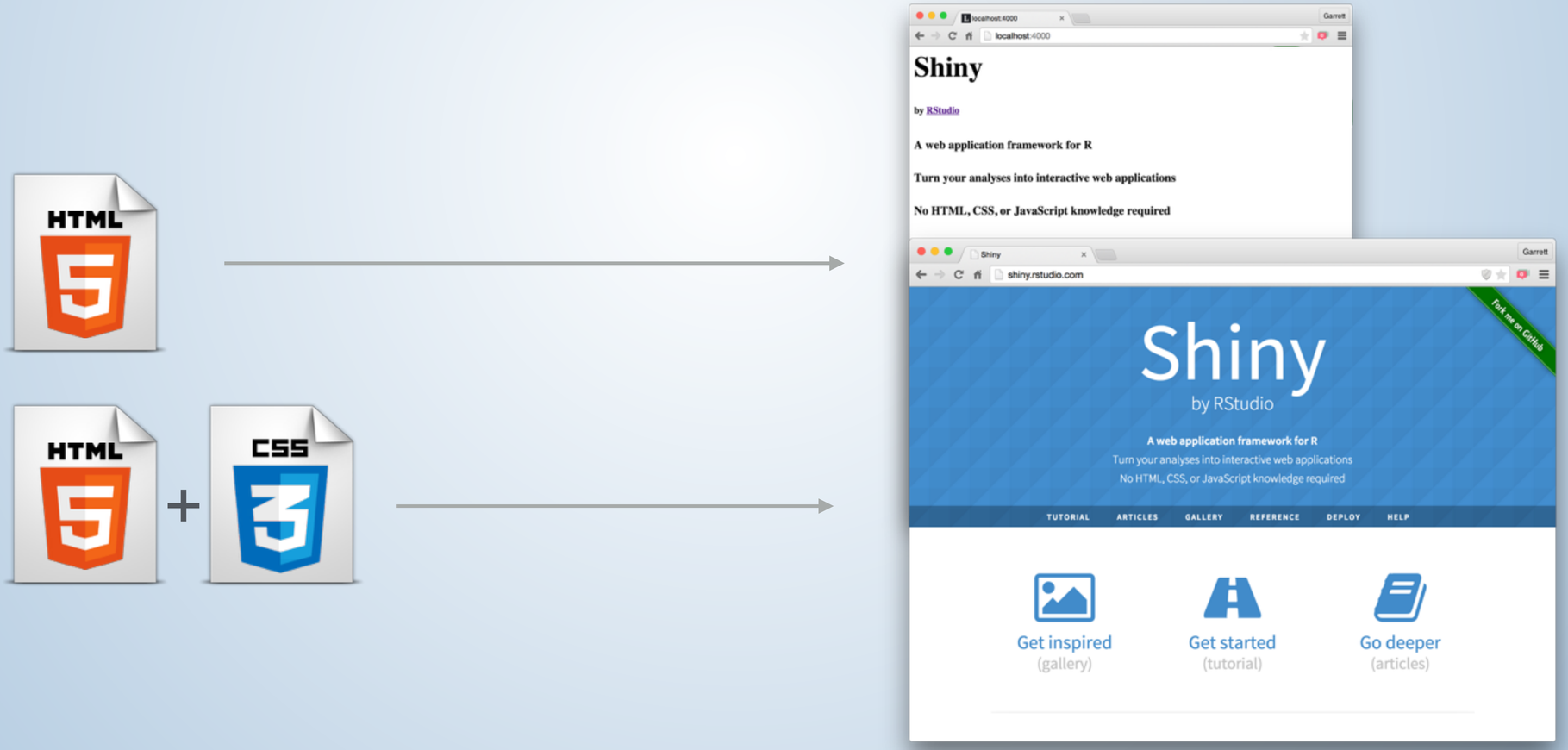
span

But there are ~600 span tags

CSS
selectors

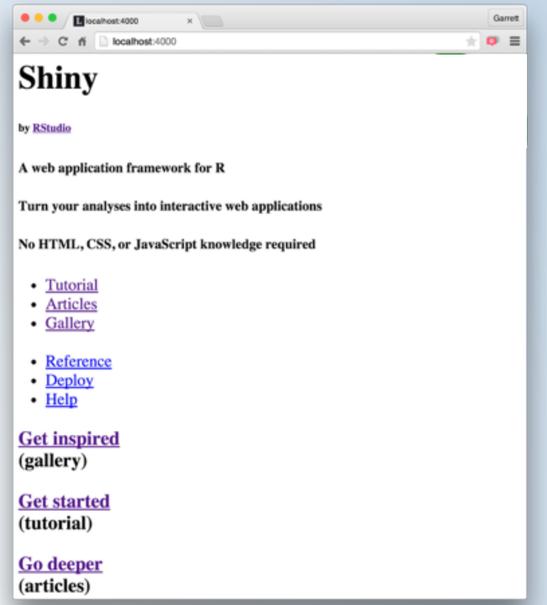
CSS (Review)

Cascading Style Sheets (CSS) are a framework for customizing the appearance of elements in a web page.

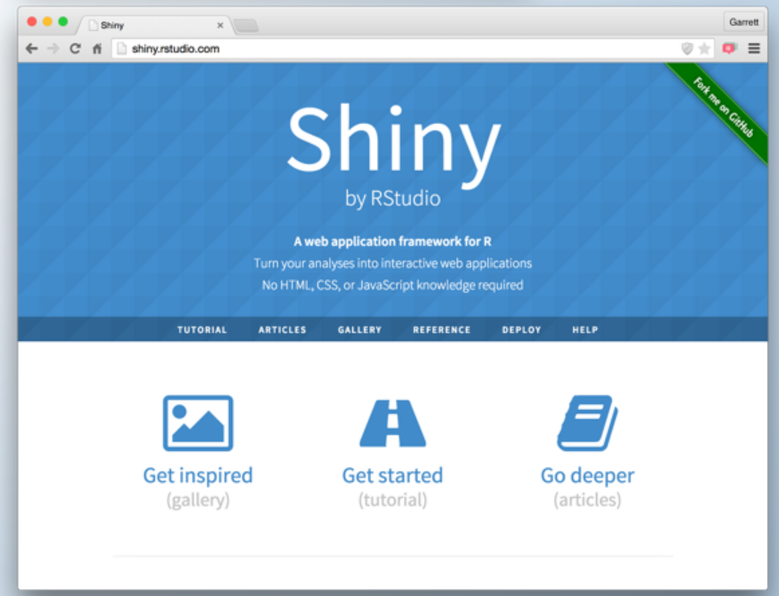


CSS (Review)


HTML


localhost:4000
Garrett
localhost:4000
Shiny
by RStudio
A web application framework for R
Turn your analyses into interactive web applications
No HTML, CSS, or JavaScript knowledge required
Tutorial
Articles
Gallery
Reference
Deploy
Help
Get inspired
(gallery)
Get started
(tutorial)
Go deeper
(articles)


HTML
CSS


Shiny
Garrett
shiny.rstudio.com
Fork me on GitHub
Shiny
by RStudio
A web application framework for R
Turn your analyses into interactive web applications
No HTML, CSS, or JavaScript knowledge required
TUTORIAL
ARTICLES
GALLERY
REFERENCE
DEPLOY
HELP
Get inspired
(gallery)
Get started
(tutorial)
Go deeper
(articles)

CSS (Review)


```
span {
  color: #ffffff;
}

.num {
  color: #a8660d;
}

table.data {
  width: auto;
}

#firstname {
  background-color: yellow;
}
```

selector
styling

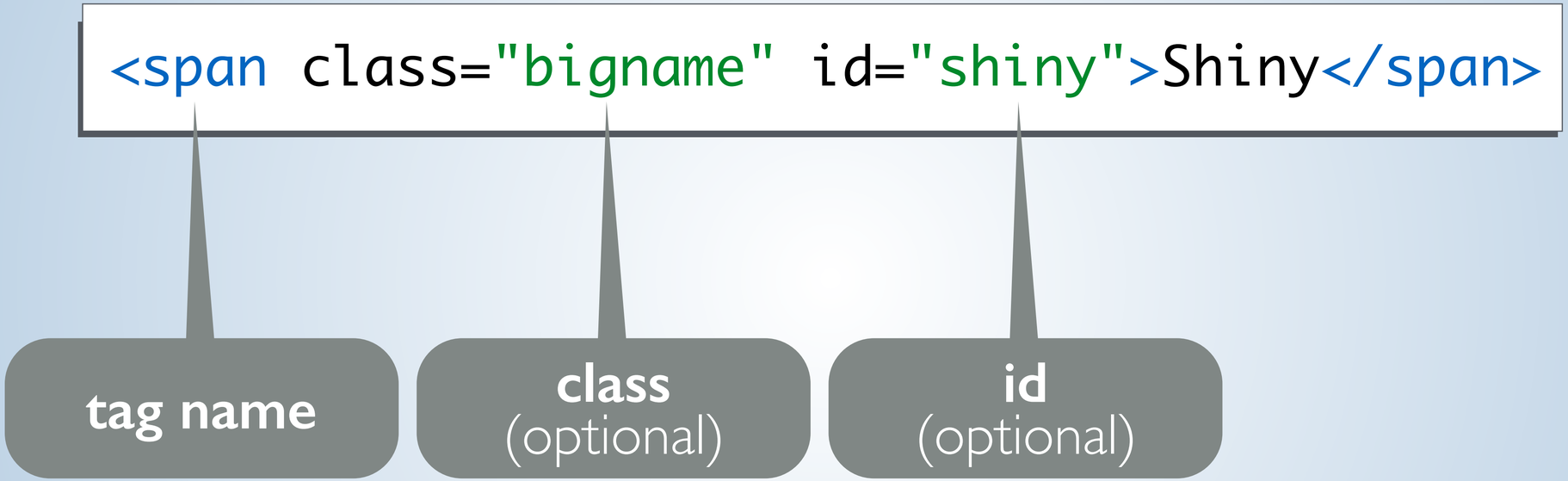
selector
styling

selector
styling

selector
styling

CSS (Review)

A CSS script describes an element by its tag, class, and/or ID.



CSS (Review)

A CSS script describes an element by its tag, class, and/or ID.

```
<span class="bigname" id="shiny">Shiny</span>
```

```
span
```

CSS selector for **ALL** elements with:

- the **span tag**

CSS (Review)

A CSS script describes an element by its tag, class, and/or ID.

```
<span class="bigname" id="shiny">Shiny</span>
```

```
.bigname
```

CSS selector for **ALL** elements with:

- the **bigname class**

CSS (Review)

A CSS script describes an element by its tag, class, and/or ID.

```
<span class="bigname" id="shiny">Shiny</span>
```

```
span.bigname
```

CSS selector for **ALL** elements with:

- the **span tag**

AND

- the **bigname class**

CSS (Review)

A CSS script describes an element by its tag, class, and/or ID.

```
<span class="bigname" id="shiny">Shiny</span>
```

```
#shiny
```

CSS selector for **ALL** elements with:

- the **shiny id**

CSS (Review)


```
span {
  color: #ffffff;
}

.num {
  color: #a8660d;
}

table.data {
  width: auto;
}

#firstname {
  background-color: yellow;
}
```

selector
styling

selector
styling

selector
styling

selector
styling

CSS (Review)

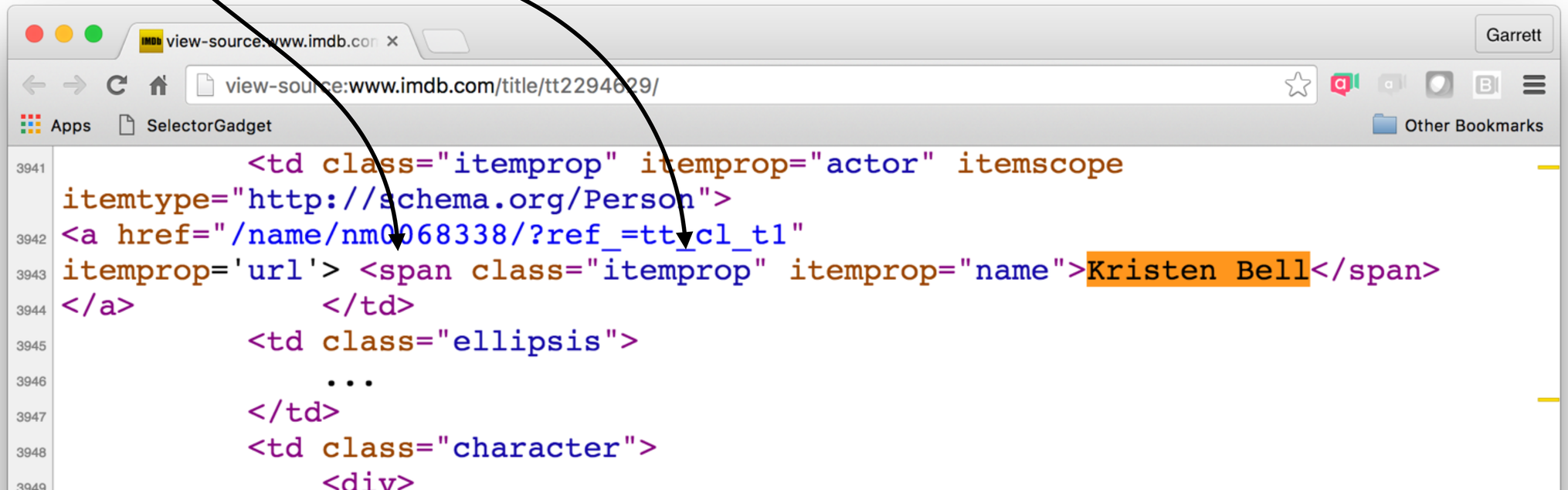
Prefix	Matches
none	tag
.	class
#	id

Which CSS identifiers are associated with Kristen Bell's name?

Which CSS identifiers are associated with Kristen Bell's name?

span (the element)

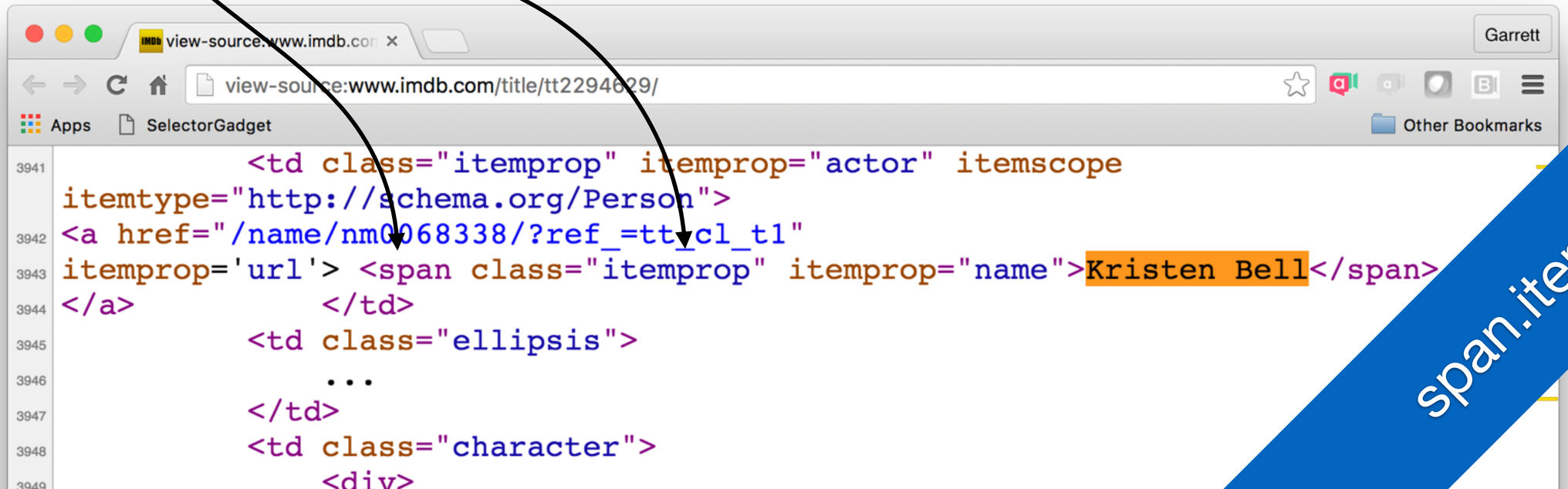
itemprop (the class)



Which CSS identifiers are associated with Kristen Bell's name?

span (the element)

itemprop (the class)



span.itemprop

Web

Recap

1. Extract information from the HTML DOM
2. Use the structure of HTML and CSS to find the information

rvest

rvest

A package that makes it easy to extract info from a webpage.

```
install.packages("rvest")
```

** This will also install xml2, a package that rvest relies on.*

Basic Workflow

1. Download the HTML and turn it into an XML file with read_html()

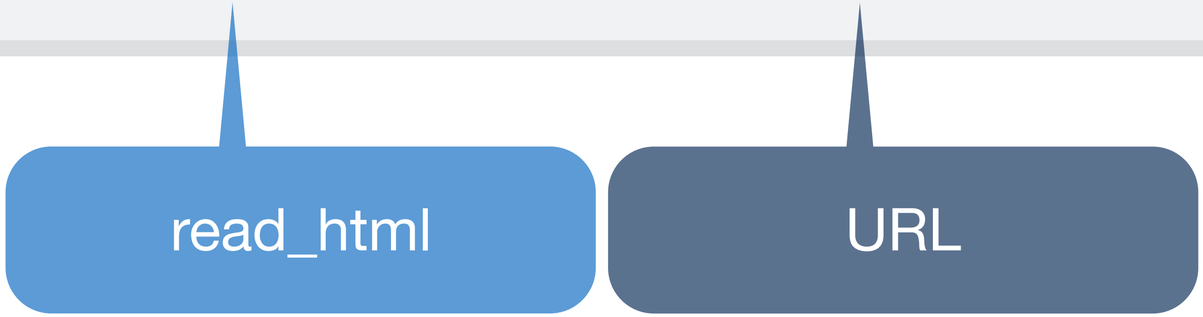
2. Extract specific nodes with html_nodes()

3. Extract content from nodes with html_text(), html_name(), html_attrs(), html_children(), html_table()

Basic Workflow

1. Download the HTML and turn it into an XML file with **read_html()**

```
library(rvest)
frozen <- read_html("http://www.imdb.com/title/tt2294629/")
```



** read_html() comes in the xml2 package loaded with rvest*

Basic Workflow

1. Download the HTML and turn it into an XML file with **read_html()**

2. Extract specific nodes with **html_nodes()**

```
cast <- html_nodes(frozen, "span.itemprop")
```

Basic Workflow

1. Download the HTML and turn it into an XML file with **read_html()**

2. Extract specific nodes with **html_nodes()**

3. Extract content from nodes with **html_text(), html_name(), html_attrs(), html_children(), html_table()**

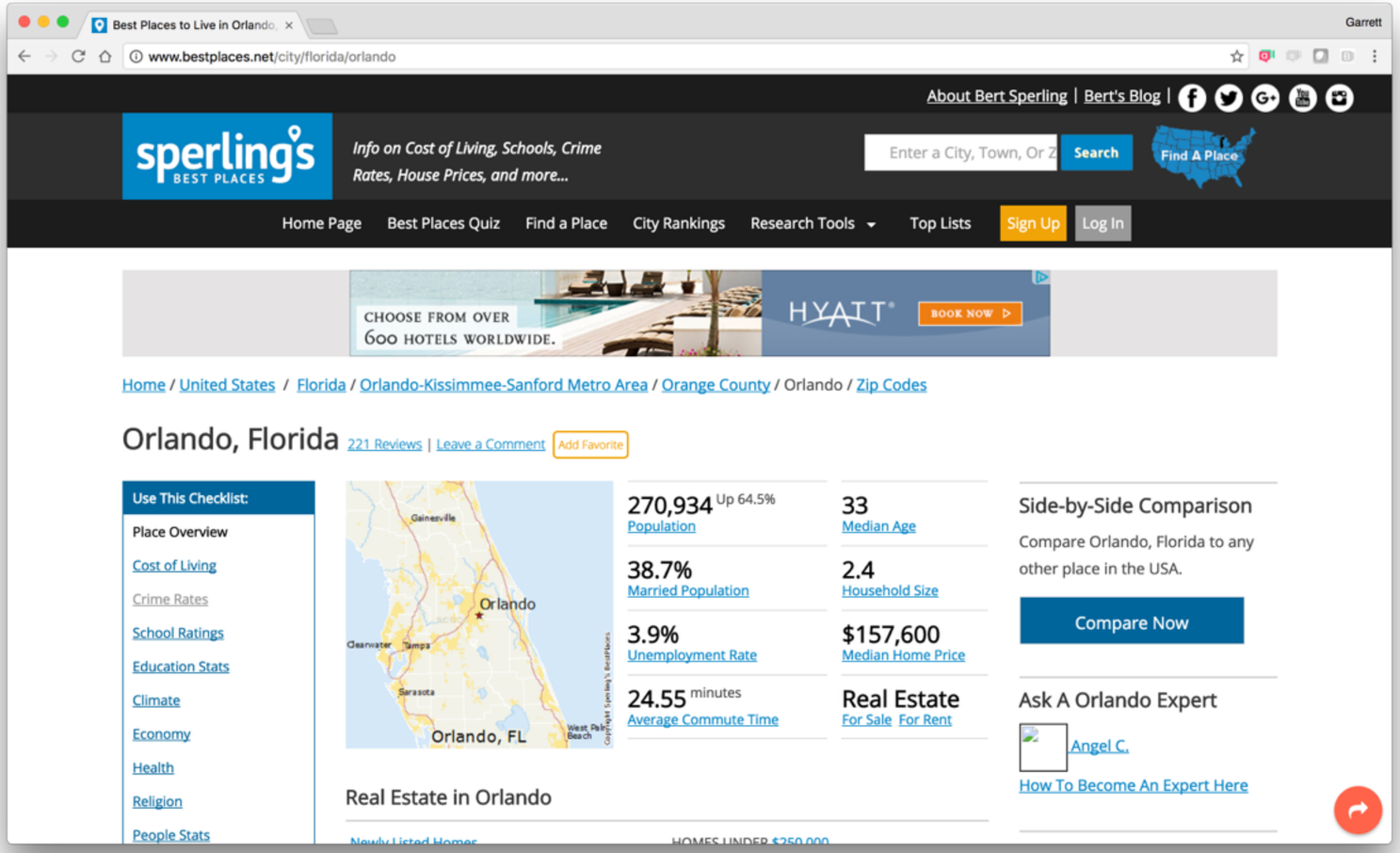
```
library(rvest)
frozen <- read_html("http://www.imdb.com/title/tt2294629/")
cast <- html_nodes(frozen, "span.itemprop")
html_text(cast)
```

```
##  [1] "Animation"                    "Adventure"
##  [3] "Comedy"                       "Chris Buck"
##  [5] "Jennifer Lee"                 "Jennifer Lee"
##  [7] "Hans Christian Andersen"      "Kristen Bell"
##  [9] "Idina Menzel"                 "Jonathan Groff"
## [11] "Kristen Bell"                 "Idina Menzel"
## [13] "Jonathan Groff"               "Josh Gad"
## [15] "Santino Fontana"              "Alan Tudyk"
## [17] "Ciarán Hinds"                 "Chris Williams"
## [19] "Stephen J. Anderson"          "Maia Wilson"
## [21] "Edie McClurg"                 "Robert Pine"
## [23] "Maurice LaMarche"             "Livvy Stubenrauch"
## [25] "Eva Bella"                    "snowman"
## [27] "sister love"                  "sister sister relationship"
## [29] "magic"                        "snow"
## [31] "Walt Disney Animation Studios" "Walt Disney Pictures"
```

We've scraped too much info

tables

www.bestplaces.net



Tables

Use html_table() to scrape whole tables of data as a data frame.

```
url <- "http://www.bestplaces.net/climate/city/florida/orlando"
orlando <- read_html(url)
tables <- html_nodes(orlando, css = "table")
html_table(tables, header = TRUE)[[2]]
```

selectorGadget

```
library(rvest)
frozen <- read_html("http://www.imdb.com/title/tt2294629/")
cast <- html_nodes(frozen, "span.itemprop")
html_text(cast)
```

```
##  [1] "Animation"                    "Adventure"
##  [3] "Comedy"                       "Chris Buck"
##  [5] "Jennifer Lee"                 "Jennifer Lee"
##  [7] "Hans Christian Andersen"      "Kristen Bell"
##  [9] "Idina Menzel"                 "Jonathan Groff"
## [11] "Kristen Bell"                 "Idina Menzel"
## [13] "Jonathan Groff"               "Josh Gad"
## [15] "Santino Fontana"              "Alan Tudyk"
## [17] "Ciarán Hinds"                 "Chris Williams"
## [19] "Stephen J. Anderson"          "Maia Wilson"
## [21] "Edie McClurg"                 "Robert Pine"
## [23] "Maurice LaMarche"             "Livvy Stubenrauch"
## [25] "Eva Bella"                    "snowman"
## [27] "sister love"                  "sister sister relationship"
## [29] "magic"                        "snow"
## [31] "Walt Disney Animation Studios" "Walt Disney Pictures"
```

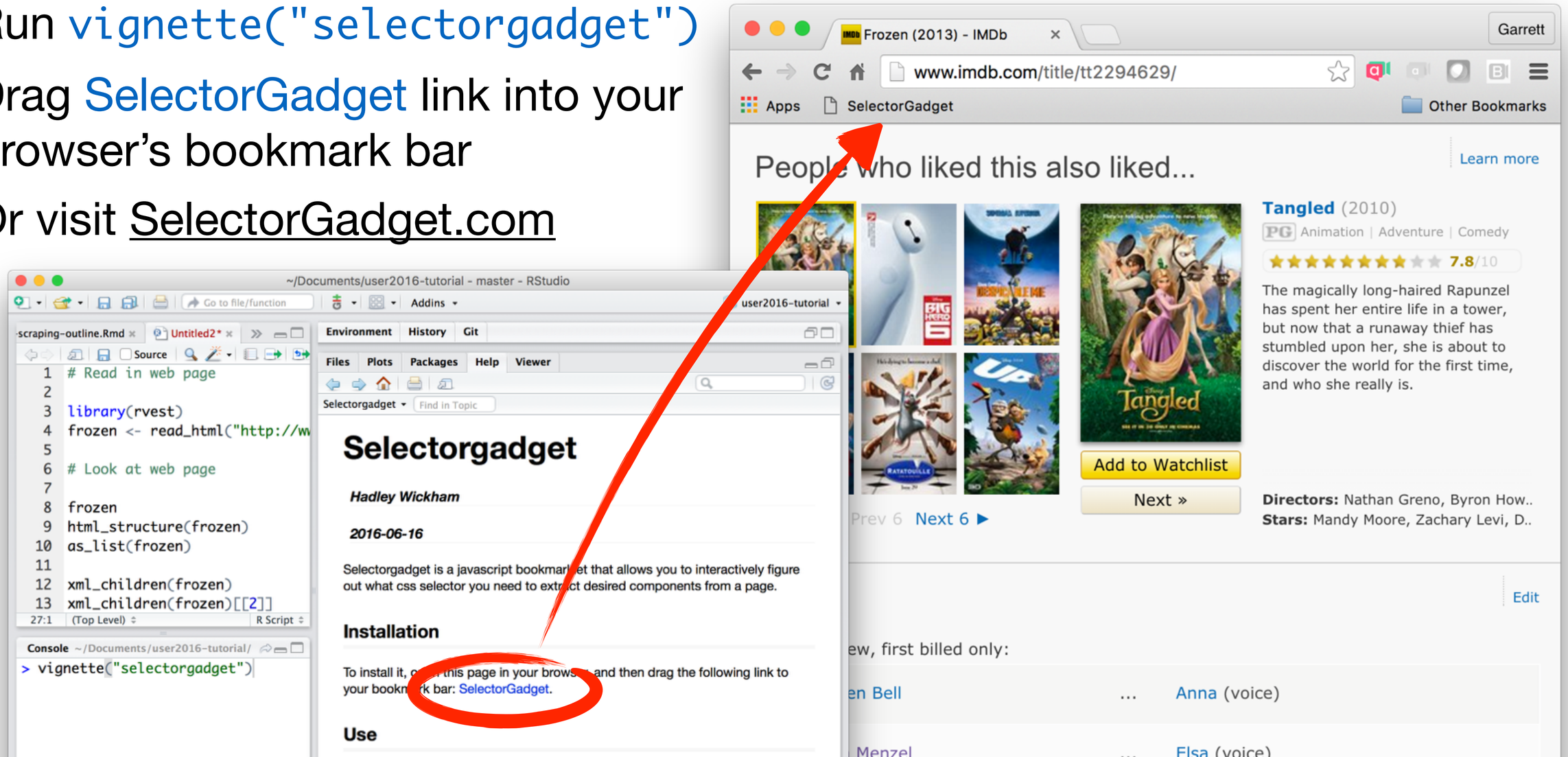
We've scraped too much info

selectorGadget

A GUI tool to identify CSS selector combinations

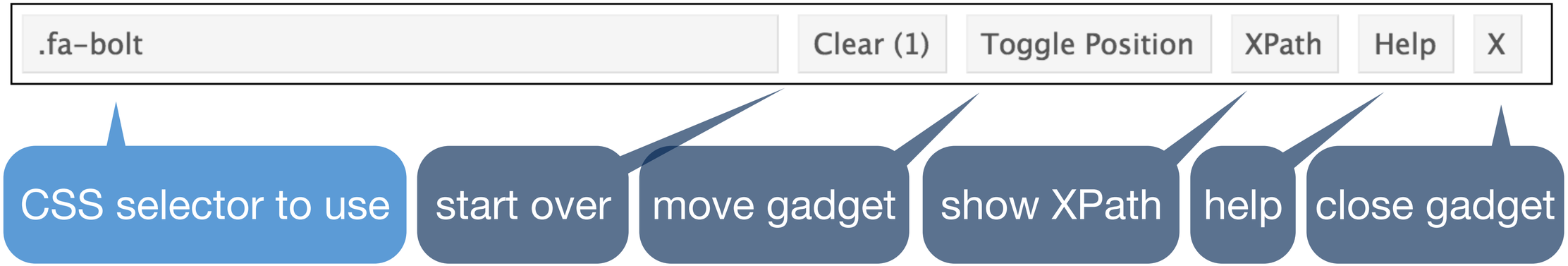
To Install

1. Run `vignette("selectorgadget")`
2. Drag SelectorGadget link into your browser's bookmark bar
3. Or visit SelectorGadget.com



To Use

1. **Navigate** to a webpage
2. **Open** the SelectorGadget bookmark
3. **Click** on item to scrape
4. **Click** on yellow items You do not want to scrape
5. **Click** on additional items that you do want to scrape
6. **Copy** selector to use with html_nodes()


.fa-bolt
Clear (1)
Toggle Position
XPath
Help
X
CSS selector to use
start over
move gadget
show XPath
help
close gadget

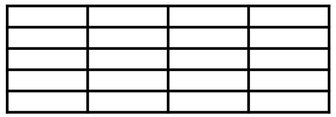
```
cast2 <- html_nodes(frozen, "#titleCast span.itemprop")
html_text(cast2)

cast3 <- html_nodes(frozen, ".itemprop .itemprop")
html_text(cast3)
```

Recap

Pull from HTML, use HTML/CSS structure

rvest

read_html() ➔ **html_nodes()** ➔ **html_text()**, etc.



html_table() for tables

selectorGadget for finding useful selector combinations

Thank You