Getting Data from the Internet

Getting Data

Image Created by Freepik


HTTP
httr
R package

HTTP

Hypertest Transfer Protocol

Allows messages to be sent on the Internet!

httr

R package

Enables YOU to work with these data in R!

2. request goes to the website

Photo by rawpixel on Unsplash

1. type URL

3. website is displayed

Photo by rawpixel on Unsplash

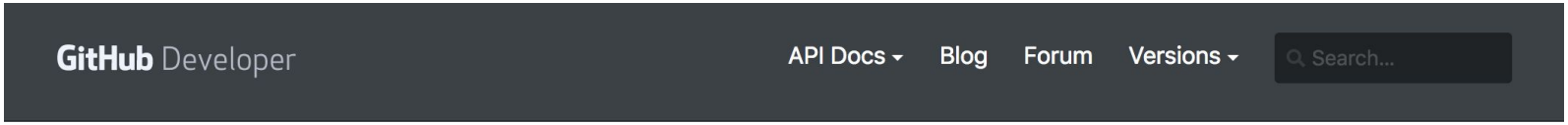
2. **API endpoint**
interprets request

```
GET(url = URL)
```

```
> api_response
Response [https://raw.githubusercontent.com/fivethirtyeight/data/master/steak-survey/steak-risk-survey.csv]
  Date: 2018-07-06 18:09
  Status: 200
  Content-Type: text/plain; charset=utf-8
  Size: 62.7 kB
RespondentID,"Consider the ...
,Response,Response,Response...
3237565956,Lottery B,,,,,,,...
3234982343,Lottery A,No,Yes...
3234973379,Lottery A,No,Yes...
3234972383,Lottery B,Yes,Ye...
3234958833,Lottery B,No,Yes...
3234955240,Lottery A,No,No,...
3234955097,Lottery A,No,Yes...
3234955010,Lottery A,No,Yes...
...
```

1. type **API request**

3. Get **API response**



REST API v3

Reference Guides Libraries

Overview

This describes the resources that make up the official GitHub REST API v3. If you have any problems or requests please contact GitHub support.

i. Current version
ii. Schema
iii. Authentication
iv. Parameters
v. Root endpoint
vi. GraphQL global node IDs
vii. Client errors
viii. HTTP redirects
ix. HTTP verbs
x. Hypermedia
xi. Pagination
xii. Rate limiting
xiii. User agent required
xiv. Conditional requests

https://developer.github.com/v3

```r
## load package
library(httr)
library(dplyr)

## Save GitHub username as variable
username <- 'janeeverydaydoe'

## Save base endpoint as variable
url_git <- 'https://api.github.com/'

## Construct API request
api_response <- GET(url = paste0(url_git,
'users/', username, '/repos'))
```

API endpoint

API request

```
## See variables in response
names(api_response)
```

```
> names(api_response)
 [1] "url"         "status_code" "headers"     "all_headers" "cookies"     "content"
 [7] "date"        "times"       "request"     "handle"
```

What API request from `httr` returns

```r
## Check Status Code of request
api_response$status_code

## Extract content from API response
repo_content <- content(api_response)
```

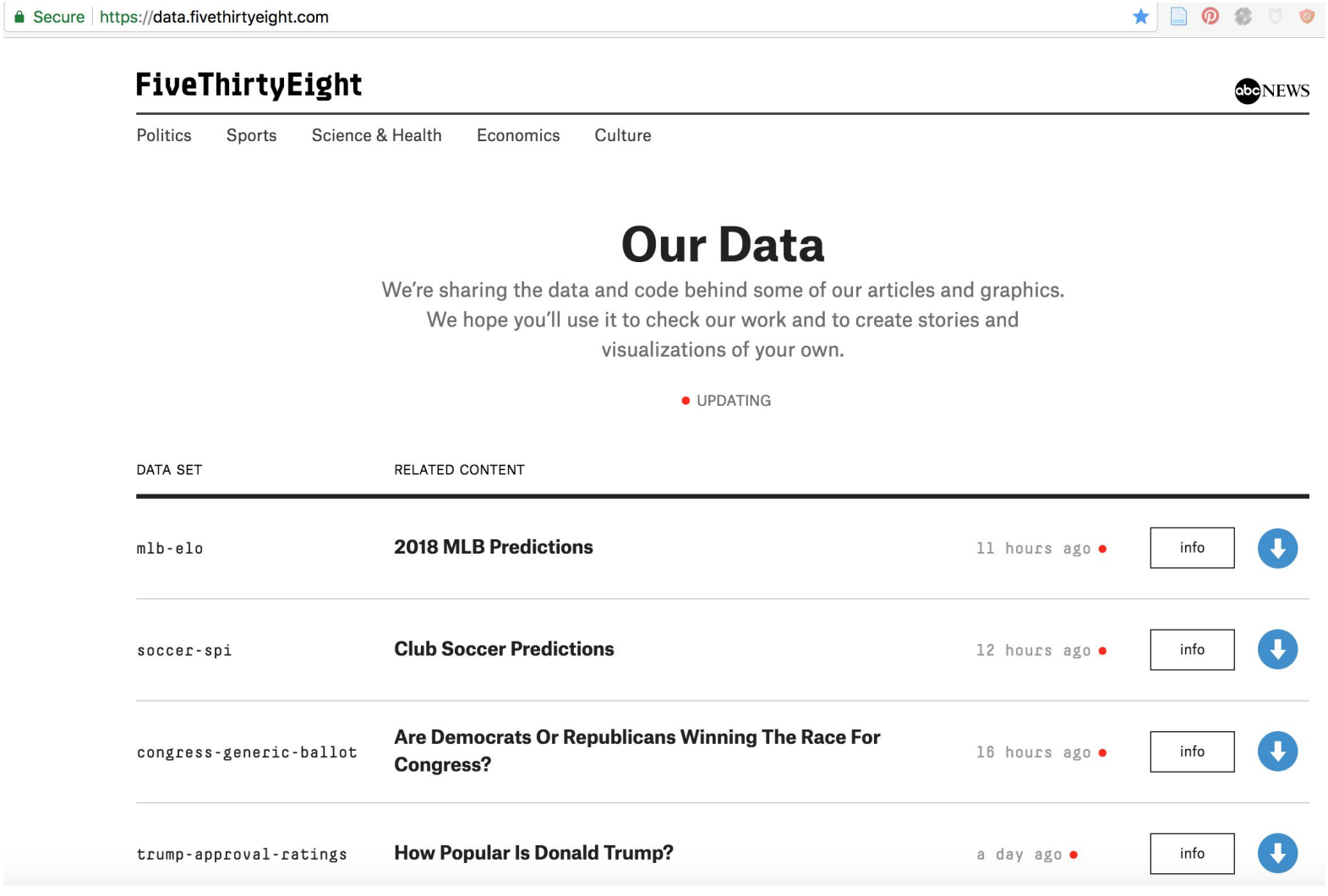
```
> api_response$status_code
[1] 200

> repo_content <- content(api_response)
```

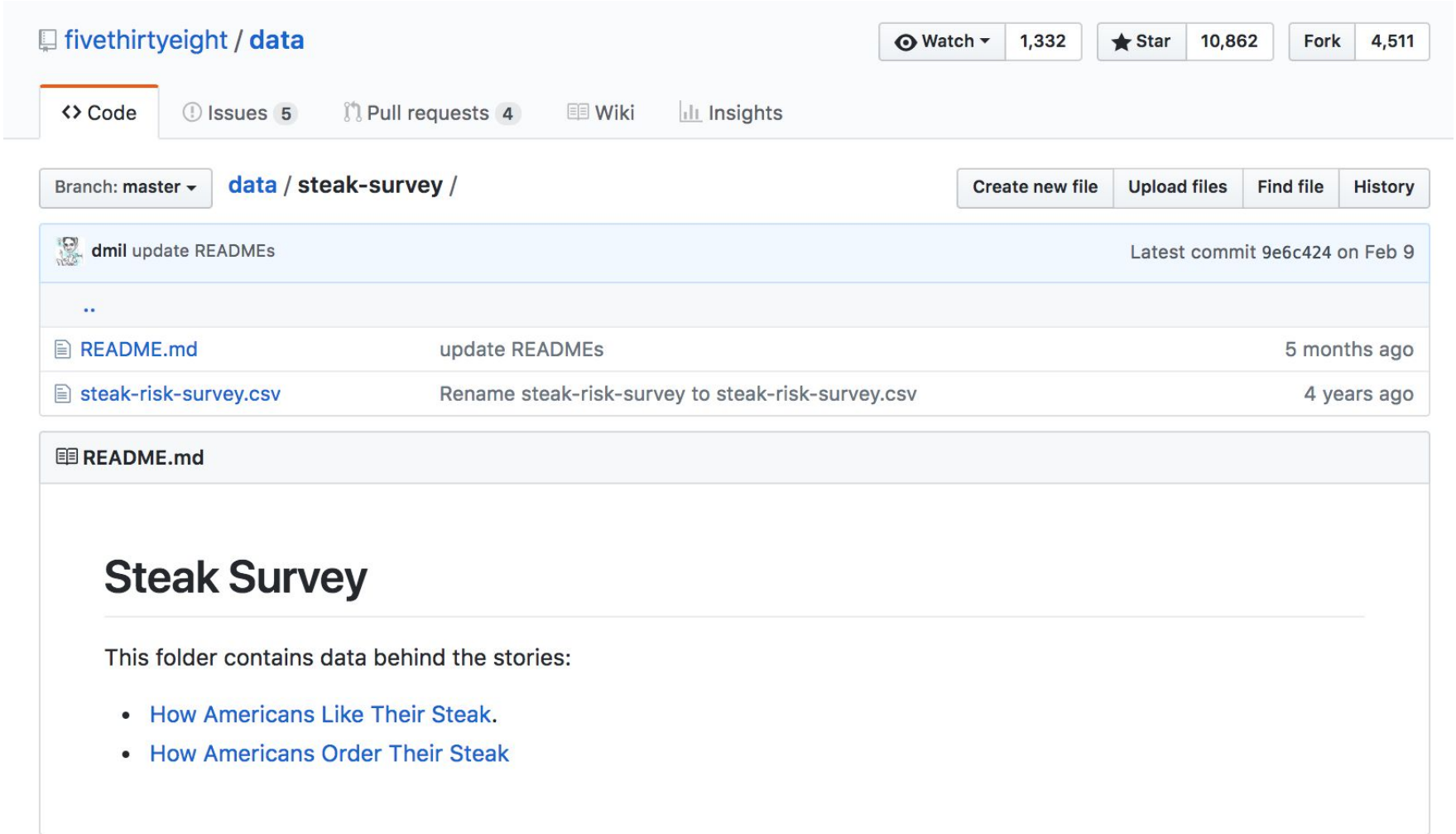
'200' means request was successful!

`content()` extracts contents from API request

```
## function to get name and URL for each repo
lapply(repo_content, function(x) {
  df <- data_frame(repo = x$name,
                   address = x$html_url)}) %>%
  bind_rows()
```

```
> lapply(repo_content, function(x) {
+     df <- data_frame(repo = x$name,
+                      address = x$html_url)}) %>%
+     bind_rows()
# A tibble: 6 x 2
  repo                              address
  <chr>                             <chr>
1 first_project                     https://github.com/JaneEverydayDoe/first_project
2 hello-world                       https://github.com/JaneEverydayDoe/hello-world
3 janeeverydaydoe.github.com        https://github.com/JaneEverydayDoe/janeeverydaydoe.g…
4 my_first_project                  https://github.com/JaneEverydayDoe/my_first_project
5 newproject                        https://github.com/JaneEverydayDoe/newproject
6 Temporary_add_to_version_control https://github.com/JaneEverydayDoe/Temporary_add_to_…
```


Secure | https://data.fivethirtyeight.com
FiveThirtyEight
ABC NEWS
Politics Sports Science & Health Economics Culture
Our Data
We're sharing the data and code behind some of our articles and graphics. We hope you'll use it to check our work and to create stories and visualizations of your own.
UPDATING
DATA SET
RELATED CONTENT
mlb-elo
2018 MLB Predictions
11 hours ago
info
soccer-spi
Club Soccer Predictions
12 hours ago
info
congress-generic-ballot
Are Democrats Or Republicans Winning The Race For Congress?
16 hours ago
info
trump-approval-ratings
How Popular Is Donald Trump?
a day ago
info



https://github.com/fivethirtyeight/data/tree/master/steak-survey

```
> ## Make API request
> api_response <- GET(url = "https://raw.githubusercontent.com/fivethirtyeight/data/master/steak-survey/steak-risk-survey.csv")
>
> ## Extract content from API response
> df_steak <- content(api_response, type="text/csv")
No encoding supplied: defaulting to UTF-8.
Parsed with column specification:
cols(
  RespondentID = col_double(),
  `Consider the following hypothetical situations: <br>In Lottery A, you have a 50% chance of success, with a payout of $100. <br>In Lottery B, you have a 90% chance of success, with a pay out of $20. <br><br>Assuming you have $10 to bet, would you play Lottery A or Lottery B?` = col_character(),
  `Do you ever smoke cigarettes?` = col_character(),
  `Do you ever drink alcohol?` = col_character(),
  `Do you ever gamble?` = col_character(),
  `Have you ever been skydiving?` = col_character(),
  `Do you ever drive above the speed limit?` = col_character(),
  `Have you ever cheated on your significant other?` = col_character(),
  `Do you eat steak?` = col_character(),
  `How do you like your steak prepared?` = col_character(),
  Gender = col_character(),
  Age = col_character(),
  `Household Income` = col_character(),
  Education = col_character(),
  `Location (Census Region)` = col_character()
)
```

`GET()` makes API request

`content()` extracts information

```
myapp = oauth_app("twitter",
                  key = "yourConsumerKeyHere",
                  secret = "yourConsumerSecretHere")
sig = sign_oauth1.0(myapp,
                    token = "yourTokenHere",
                    token_secret = "yourTokenSecretHere")
homeTL =
GET("https://api.twitter.com/1.1/statuses/home_timeline.js
on", sig)
```

Photo by rawpixel on Unsplash

https://github.com/hadley/rvest

Chromebook Data Science Curriculum

There are currently 12 courses that are offered in the Chromebook Data Science Curriculum.

Class	Course Description	Leanpub Link
Introduction to Chromebook Data Science	This is the first class in the Chromebook Data Science series. Data science is one of the most exciting and fastest growing careers in the world. The goal of this series is to help people with no background and limited resources transition into data science. The only pre-requisites are a computer with a web browser and the ability to type and follow instructions. We guide you through the rest.	Course 0
How to Use A Chromebook	This course will introduce you to using a Chromebook. The Introduction and Setup course might sound simple, but it will set up the infrastructure for success with the later, more challenging courses.	Course 1
Google and the Cloud	The Google and the Cloud course introduces using Google's in-built apps, which form the fundamental backbone of a Chromebook. We'll go step by step through the process to integrating these apps together to form your productivity workflow.	Course 2
Organizing Data Science	Projects are central to the role of any data scientist. These lessons will	Course 3

Different parts of the webpage have different HTML tags

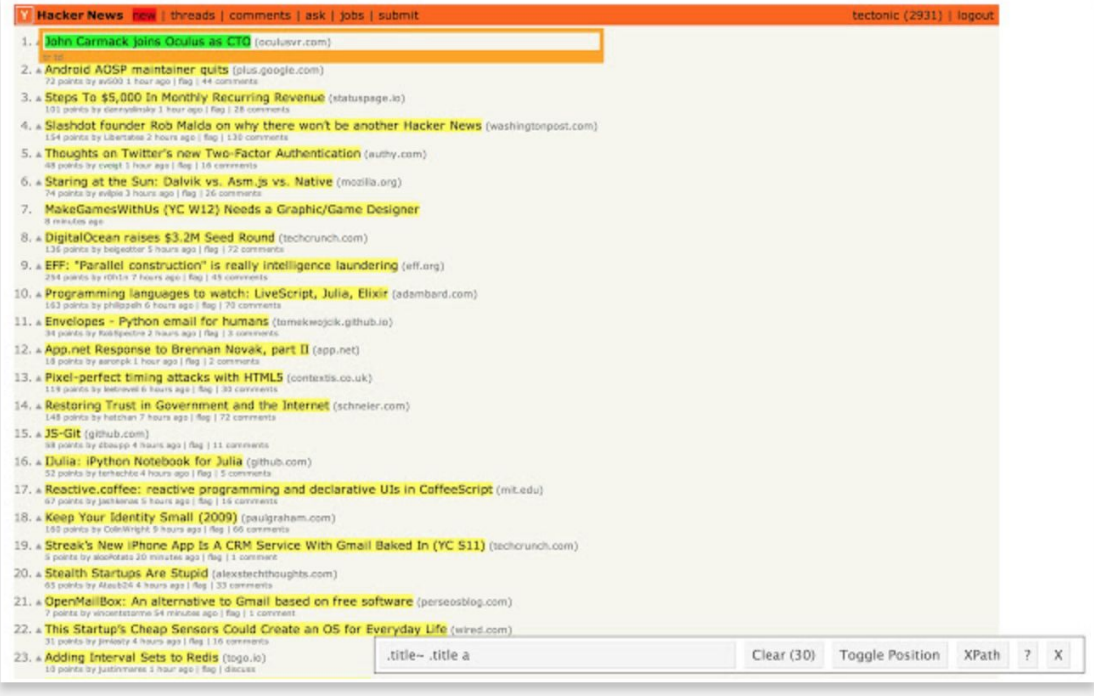
SelectorGadget

offered by selectorgadget.com

★★★★½ (68) | Developer Tools | 78,479 users

+ ADD TO CHROME

OVERVIEW | REVIEWS | SUPPORT | RELATED



Compatible with your device

Easy, powerful CSS Selector generation.

Selector Gadget is an open source Chrome Extension that makes CSS selector generation and discovery on complicated sites a breeze.

After having installed the extension, go to any page and launch it. A box will open in the bottom right of the website. Click on a page element that you would like your selector to match (it will turn green). SelectorGadget will then generate a minimal CSS selector for that element, and will highlight (yellow) everything that is matched

Website

Report Abuse

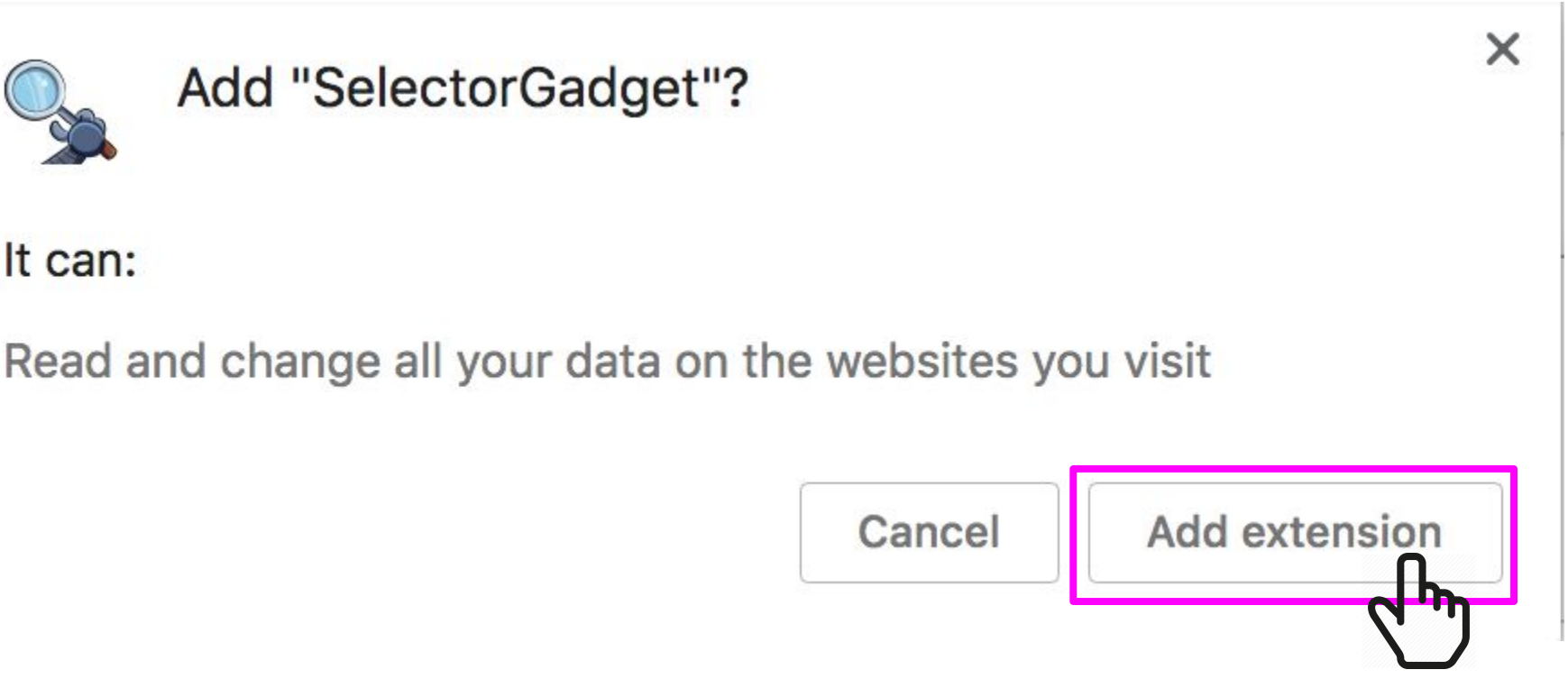
Additional Information

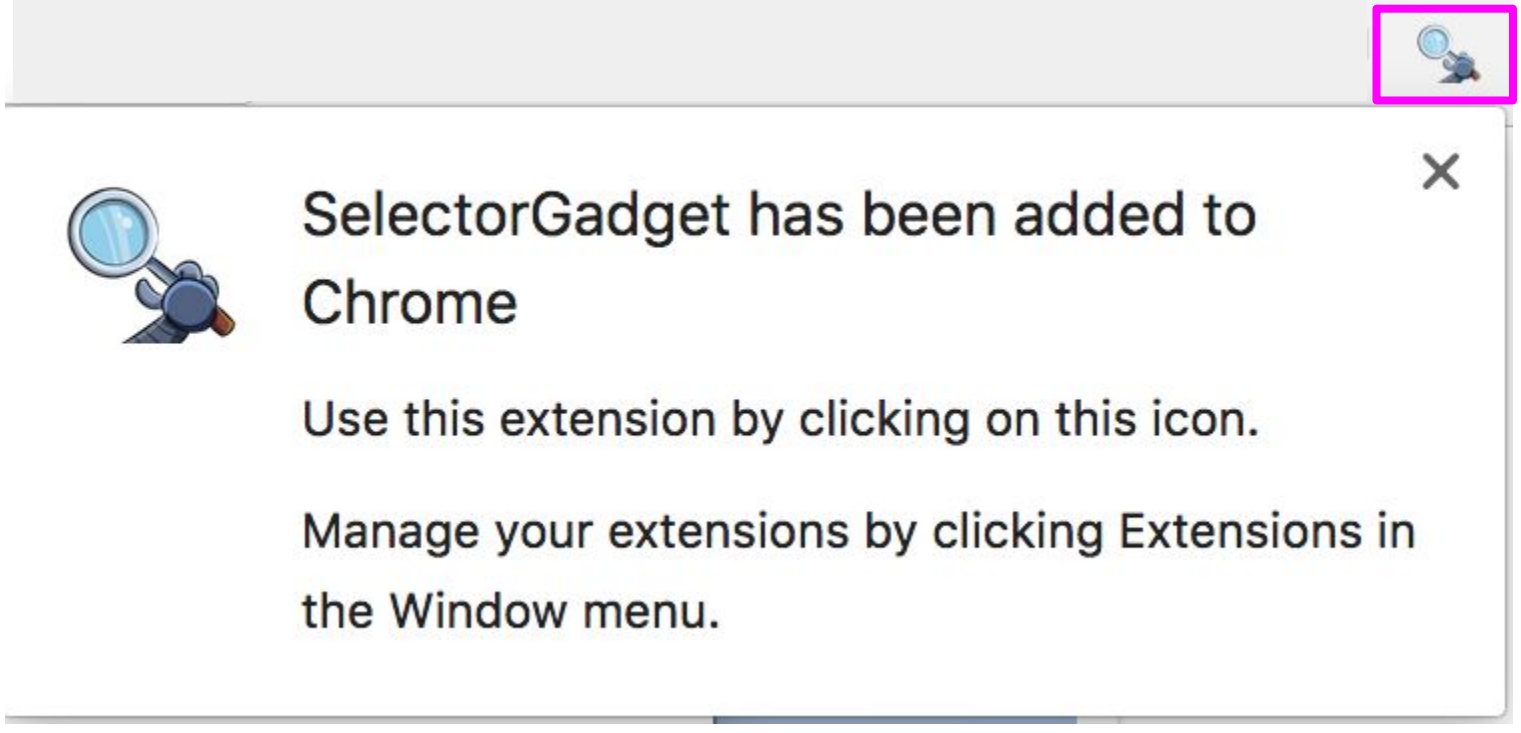
Version: 1.1

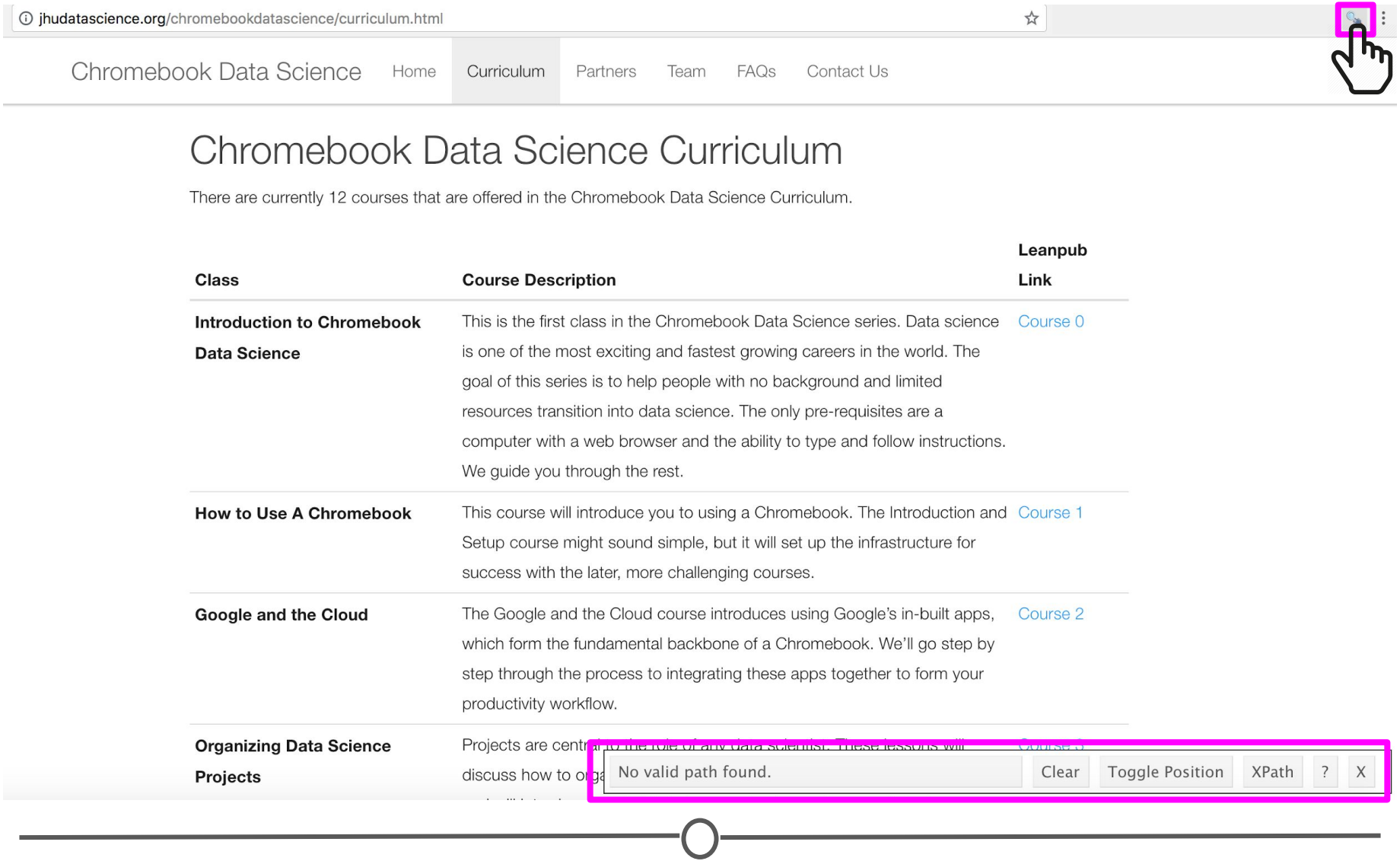
Updated: March 20, 2015

Size: 79.7KiB

Language: English


Add "SelectorGadget"?
It can:
Read and change all your data on the websites you visit
Cancel
Add extension


SelectorGadget has been added to Chrome
Use this extension by clicking on this icon.
Manage your extensions by clicking Extensions in the Window menu.


jhudatascience.org/chromebookdatascience/curriculum.html
Chromebook Data Science
Home
Curriculum
Partners
Team
FAQs
Contact Us
Chromebook Data Science Curriculum
There are currently 12 courses that are offered in the Chromebook Data Science Curriculum.
Class
Course Description
Leanpub Link
Introduction to Chromebook Data Science
This is the first class in the Chromebook Data Science series. Data science is one of the most exciting and fastest growing careers in the world. The goal of this series is to help people with no background and limited resources transition into data science. The only pre-requisites are a computer with a web browser and the ability to type and follow instructions. We guide you through the rest.
Course 0
How to Use A Chromebook
This course will introduce you to using a Chromebook. The Introduction and Setup course might sound simple, but it will set up the infrastructure for success with the later, more challenging courses.
Course 1
Google and the Cloud
The Google and the Cloud course introduces using Google's in-built apps, which form the fundamental backbone of a Chromebook. We'll go step by step through the process to integrating these apps together to form your productivity workflow.
Course 2
Organizing Data Science Projects
Projects are centr
discuss how to or
No valid path found.
Clear
Toggle Position
XPath
?
X

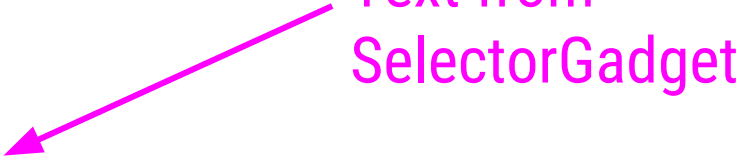
Chromebook Data Science Curriculum

There are currently 12 courses that are offered in the Chromebook Data Science Curriculum.

Class	Course Description	Leanpub Link
Introduction to Chromebook Data Science td strong	This is the first class in the Chromebook Data Science series. Data science is one of the most exciting and fastest growing careers in the world. The goal of this series is to help people with no background and limited resources transition into data science. The only pre-requisites are a computer with a web browser and the ability to type and follow instructions. We guide you through the rest.	Course 0
How to Use A Chromebook	This course will introduce you to using a Chromebook. The Introduction and Setup course might sound simple, but it will set up the infrastructure for success with the later, more challenging courses.	Course 1
Google and the Cloud	The Google and the Cloud course introduces using Google's in-built apps, which form the fundamental backbone of a Chromebook. We'll go step by step through the process to integrating these apps together to form your productivity workflow.	Course 2
Organizing Data Science Projects	Projects are central to the role of any data scientist. These lessons will discuss how t	Course 3

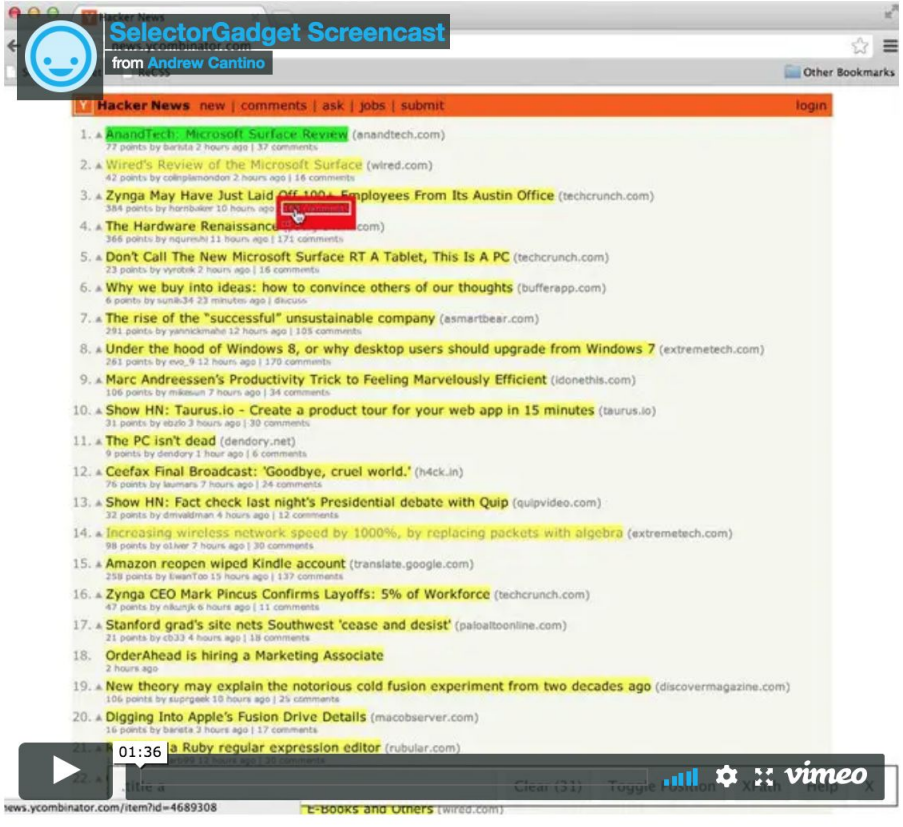
Text you'll use to specify which part of the webpage you'd like to scrape

```
## load package
library(rvest)

## provide URL
courses <-
read_html("http://jhudatascience.org/chromebook
datascience/curriculum.html")

## Get Courses
courses %>%
  html_nodes("strong") %>%
  html_text()
```



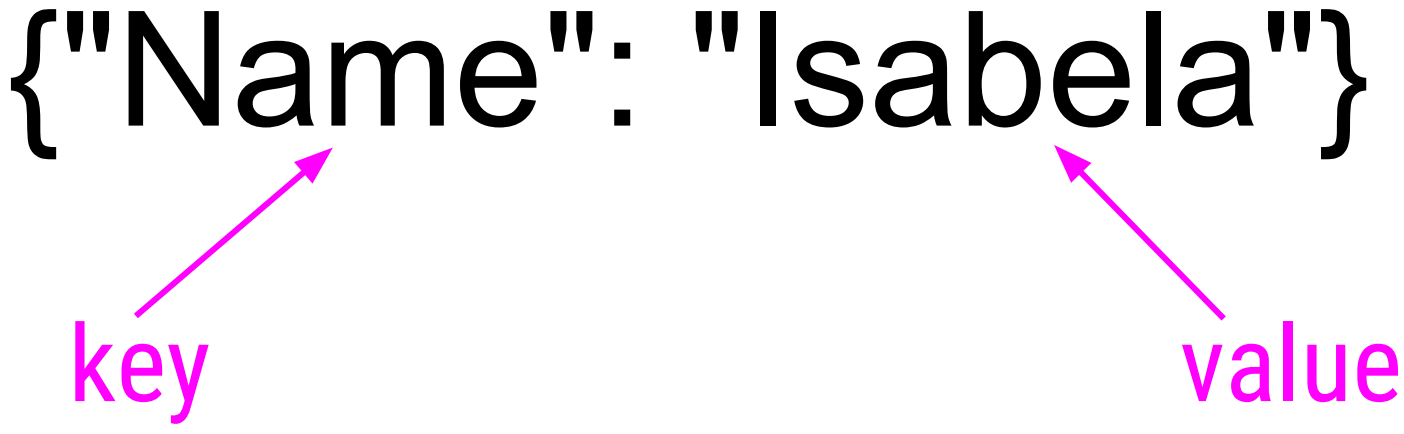
```
> courses %>%
+     html_nodes("strong") %>%
+     html_text()
 [1] "Introduction to Chromebook Data Science"
 [2] "How to Use A Chromebook"
 [3] "Google and the Cloud"
 [4] "Organizing Data Science Projects"
 [5] "Version Control"
 [6] "Introduction to R"
 [7] "Data: Tidying"
 [8] "Data Visualization"
 [9] "Getting Data"
[10] "Basics of Data Analysis"
[11] "Written and Oral Communication in DS"
[12] "Getting a job in data science"
```

SelectorGadget:
point and click CSS selectors



SelectorGadget Screencast from Andrew Cantino on Vimeo.

JSON: key-value pairs



```
library(jsonlite)

## generate a JSON object
json <-
'[
  {"Name" : "Woody", "Age" : 40, "Occupation" :
"Sherriff"},
  {"Name" : "Buzz Lightyear", "Age" : 34,
"Occupation" : "Space Ranger"},
  {"Name" : "Andy", "Occupation" : "Toy Owner"}
]'

## take a look
json
```

```
> json
[{"Name":"Woody","Age":40,"Occupation":"Sherriff"},{"Name":"Buzz Lightyear","Age":34,"Occupati
on":"Space Ranger"},{"Name":"Andy","Occupation":"Toy Owner"}]
```

```
## take JSON object and covert to a
data frame
mydf <- fromJSON(json)

## take a look
mydf
```

```
> mydf
                Name Age    Occupation
1              Woody  40      Sherriff
2 Buzz Lightyear  34 Space Ranger
3              Andy  NA     Toy Owner
```

```
## take JSON object and convert to a
data frame
json <- toJSON(mydf)
```

```
> json
[{"Name":"Woody","Age":40,"Occupation":"Sherriff"},{"Name":"Buzz Lightyear","Age":34,"Occupati
on":"Space Ranger"},{"Name":"Andy","Occupation":"Toy Owner"}]
```

Photo by rawpixel on Unsplash