Overview

Human browsing versus web scraping

HTTP overview

URL hacking

Human Browsing Versus Web Scraping

Human Versus Web Scraper

Human/Browser	Web Scraper

Human Versus Web Scraper

Human/Browser	Web Scraper
Enter a URL or click a bookmark	Set a start_url

Human Versus Web Scraper

Human/Browser	Web Scraper
Enter a URL or click a bookmark	Set a start URL
Download HTML	Download HTML

Human Versus Web Scraper

Human/Browser	Web Scraper
Enter a URL or click a bookmark	Set a start URL
Download HTML	Download HTML
Parse HTML & render	Parse HTML

Human Versus Web Scraper

Human/Browser	Web Scraper
Enter a URL or click a bookmark	Set a start URL
Download HTML	Download HTML
Parse HTML & render	Parse HTML
Review for Useful Information	Extract Useful Information

Human Versus Web Scraper

Human/Browser	Web Scraper
Enter a URL or click a bookmark	Set a start URL
Download HTML	Download HTML
Parse HTML & render	Parse HTML
Review for Useful Information	Extract Useful Information
Interpret	Transform or Aggregate

Human Versus Web Scraper

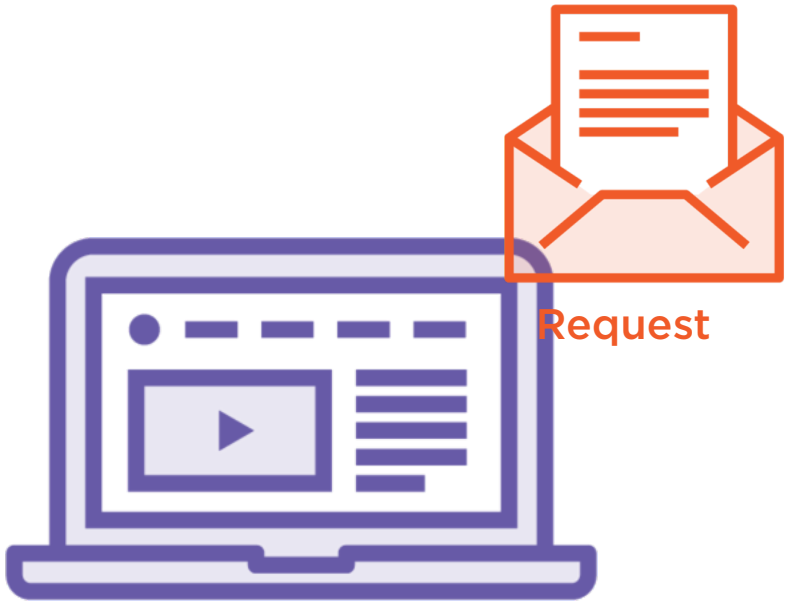
Human/Browser	Web Scraper
Enter a URL or click a bookmark	Set a start URL
Download HTML	Download HTML
Parse HTML & render	Parse HTML
Review for Useful Information	Extract Useful Information
Interpret	Transform or Aggregate
Remember the Information	Save the Data

Human Versus Web Scraper

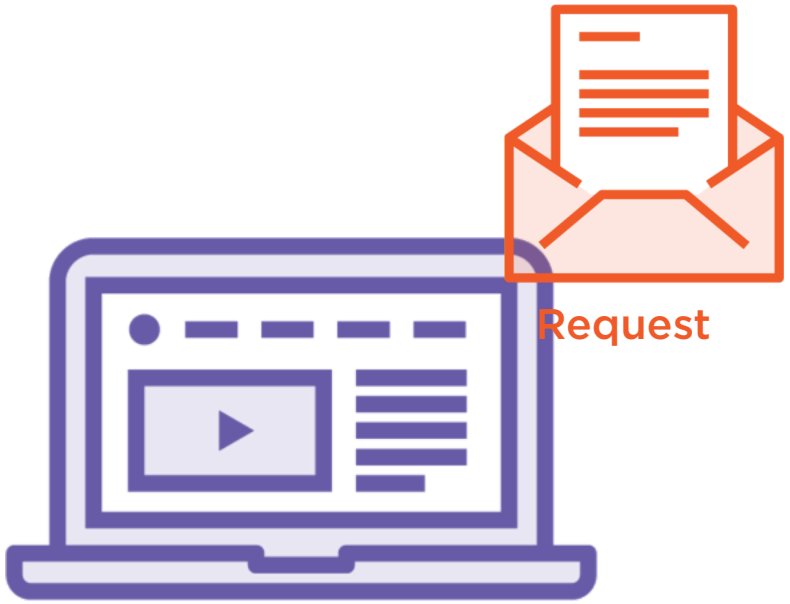
Human/Browser	Web Scraper
Enter a URL or click a bookmark	Set a start URL
Download HTML	Download HTML
Parse HTML & render	Parse HTML
Review for Useful Information	Extract Useful Information
Interpret	Transform or Aggregate
Remember the Information	Save the Data
Click a link-Enter another URL	Go the the next URL

HTTP Overview

Request - Response



Request - Response



Hyper-Text Transfer Protocol (HTTP) is the protocol that powers the web.

HTTP Request

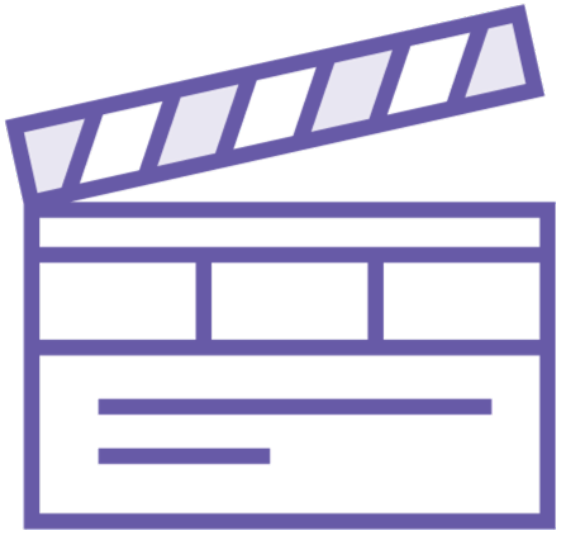
HTTP requests may include:

1 – A web address or URL

2 - A “verb”

3 - User Agent

HTTP Request: Verb



GET – Retrieves data

POST – Sends data to the server

HTTP Request: User Agent

Identifies the browser or web scraper

Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.87 Safari/537.36

URL Hacking

Used Tesla Model 3 For Sale in Lebanon, KS

Save Search

Zip Code

66952

Radius

Nation-Wide

Make

Tesla

Model

Model 3

Year

Min to Max

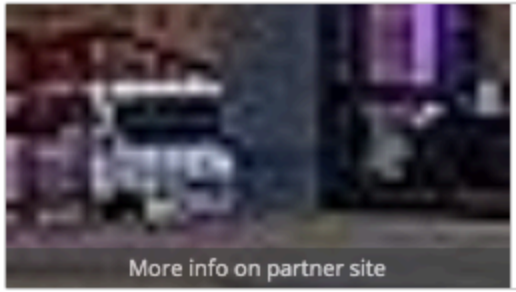
Trim

All

1-15 of 20 Used Cars Found

Best Deals First

Location:66952 X Make:Tesla X Model:Model 3 X Condition:Used X Save Search



2018 Tesla Model 3 - 22,835 mi

Ballwin, MO (435 mi) - Listed 11 days ago

$536 above market price

★★★½☆ dealer rating

1-owner , low miles , free CARFAX

$45,500

~~$45,995~~

Fair Deal

PREVIEW

SAVE

2019 Tesla Model 3 Long Range - 2,150 mi

Denver, CO (346 mi) - Listed 4 days ago

★★★★★ dealer rating

1-owner , low miles , free CARFAX

$51,995

PREVIEW

SAVE

2019 Tesla Model 3 Long Range - 2,027 mi

Denver, CO (348 mi) - Listed 2 days ago

$49,091

~~$49,690~~

PREVIEW

Used Tesla Model 3 For Sale in Lebanon, KS

Save Search

1-15 of 20 Used Cars Found

Best Deals First

Zip Code

66952

Radius

Nation-Wide

Make

Tesla

Model

Model 3

Year

Min to Max

Trim

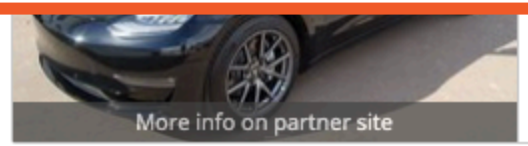
All

https://www.iseecars.com/used-cars/used-tesla-for-sale#Location=66952&Radius=all&Make=Tesla&Model=Model+3&Condition=used&_t=a&maxResults=15&sort=BestDeal&sortOrder=desc&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D

Save Search

$45,500
~~$45,995~~
Fair Deal
PREVIEW
SAVE

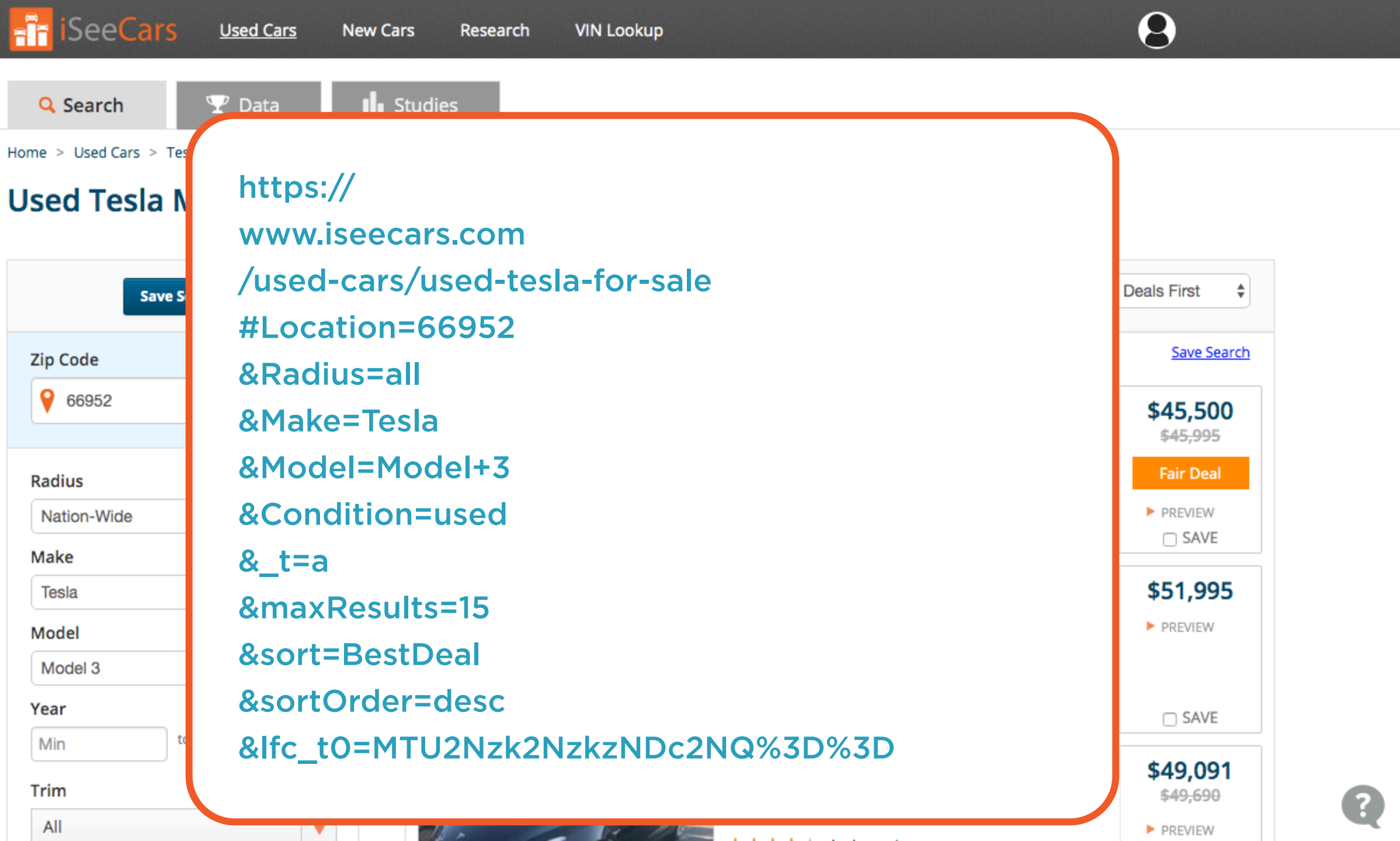
$51,995
PREVIEW
SAVE



★★★★★ dealer rating
1-owner , low miles , free CARFAX

2019 Tesla Model 3 Long Range - 2,027 mi
Denver, CO (348 mi) - Listed 2 days ago

$49,091
~~$49,690~~
PREVIEW


iSeeCars
Used Cars
New Cars
Research
VIN Lookup
Search
Data
Studies
Home > Used Cars > Tes
Used Tesla M
Save S
Zip Code
66952
Radius
Nation-Wide
Make
Tesla
Model
Model 3
Year
Min
Trim
All
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
Deals First
Save Search
$45,500
$45,995
Fair Deal
PREVIEW
SAVE
$51,995
PREVIEW
SAVE
$49,091
$49,690
PREVIEW

Scheme

iseescars.com URL

```
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

Host

iseescars.com URL

```
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

Port

iseescars.com URL

```
https://
www.iseecars.com:443
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

Path

iseescars.com URL

```
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

Query String (?)
or
URL Fragment (#)

iseescars.com URL

```
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

Query String

iseescars.com URL

```
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

Query String

iseescars.com URL

```
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

Query String

iseescars.com URL

```
https://
www.iseecars.com
/used-cars/used-tesla-for-sale
#Location=66952
&Radius=all
&Make=Tesla
&Model=Model+3
&Condition=used
&_t=a
&maxResults=15
&sort=BestDeal
&sortOrder=desc
&lfc_t0=MTU2Nzk2NzkzNDc2NQ%3D%3D
```

```
host = 'www.iseecars.com'
path = '/used-cars/used-tesla-for-sale'
location = '66952'
query_string = f'#Location={location}&Radius=all&Make=Tesla&Model=Model+3'

start_url = f'http://{host}{path}{query_string}'
```

Python URL Strings

```
import requests
start_url = ‘https://www.iseecars.com/used-cars/used-tesla-for-sale’

downloaded_page = requests.get(start_url)

print(downloaded_page.text)
```

Python Requests

Summary

Human browsing versus web scraping

HTTP protocol

URL hacking

Overview

HTML & CSS selectors

XPath

Chrome developer tools

HTML & CSS Selectors

HTML Example

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
</body>
</html>
```

Used Tesla Cars in Your Area

2017 Tesla Model 3

$34,586

A wonderful car and a great deal.

2019 Tesla Model 3

$37,938

Great deal. Low mileage.

2018 Tesla Model 3

$36,263

Bright red. Attract attention.

HTML Example

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
</body>
</html>
```

Select the Title Tag

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
```

```
css => 'title'
```

Select an H1 Tag

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
```

```
css => 'h1'
```

Select Multiple <li> Elements

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
```

```
css => 'li'
```

Select with an HTML ID

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
```

```
css => '#vin3827'
```

Select with a CSS Class

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
```

```
css => '.auto-listing'
```

Select by Parent-Child

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
```

```
css => 'ul li'
```

Combined Selectors

```
<html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1 id="main-page-title">Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <div>
                <h2 class="model">2017 Tesla Model 3</h2>
                <p class="price">$34,586</p>
                <p class="description">
                    A wonderful car and a great deal.</p>
            </div>
        </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
```

```
css => 'ul.listings li#vin3827'
```

```
example = open("example.html","r")
html = example.read()
# html = requests.get(url).text
example.close()

from bs4 import BeautifulSoup
soup = BeautifulSoup(html)
print(soup.prettify())
```

Beautiful Soup

```
example = open("example.html","r")
html = example.read()
# html = requests.get(url).text
example.close()

from bs4 import BeautifulSoup
soup = BeautifulSoup(html)
print(soup.prettify())
```

Beautiful Soup

```
soup.title
```

<title>Car Buying Web Page</title>

```
soup.title
```

<title>Car Buying Web Page</title>

```
soup.li
```

```
<li id="vin3827" class=“auto-listing">
  <div>
    <h2 class="model">2017 Tesla
          Model 3</h2>
    <p class="price">$34,586</p>
    <p class="description">A wonderful
        car and a great deal.</p>
  </div>
</li>
```

```
soup.title
```

```
<title>Car Buying Web Page</title>
```

```
soup.li
```

```
<li id="vin3827" class="auto-listing">
  <div>
    <h2 class="model">2017 Tesla
        Model 3</h2>
    <p class="price">$34,586</p>
    <p class="description">A wonderful
       car and a great deal.</p>
  </div>
</li>
```

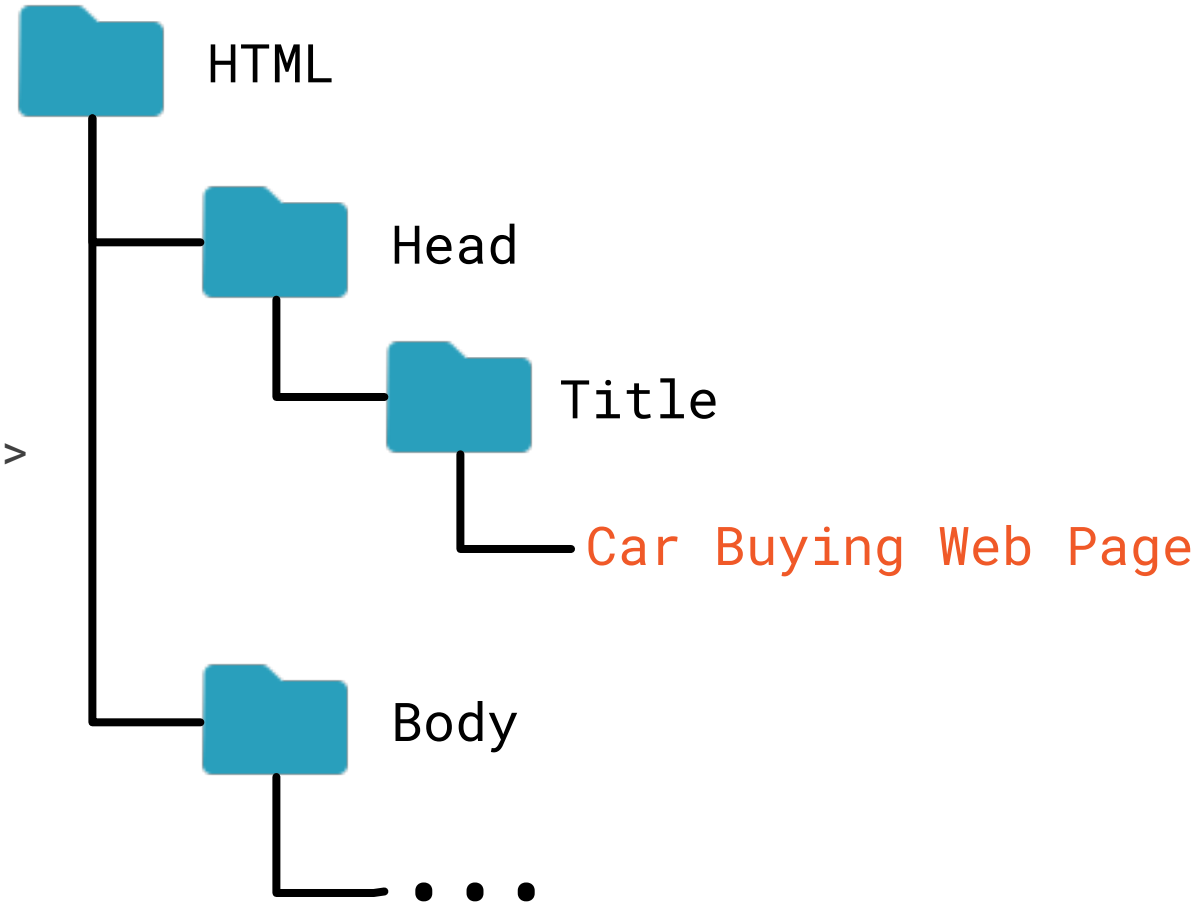
```
soup.find_all('li')
```

```
[ <li> ... </li>, <li> ... </li>, <li> ... </li> ]
```

XPath

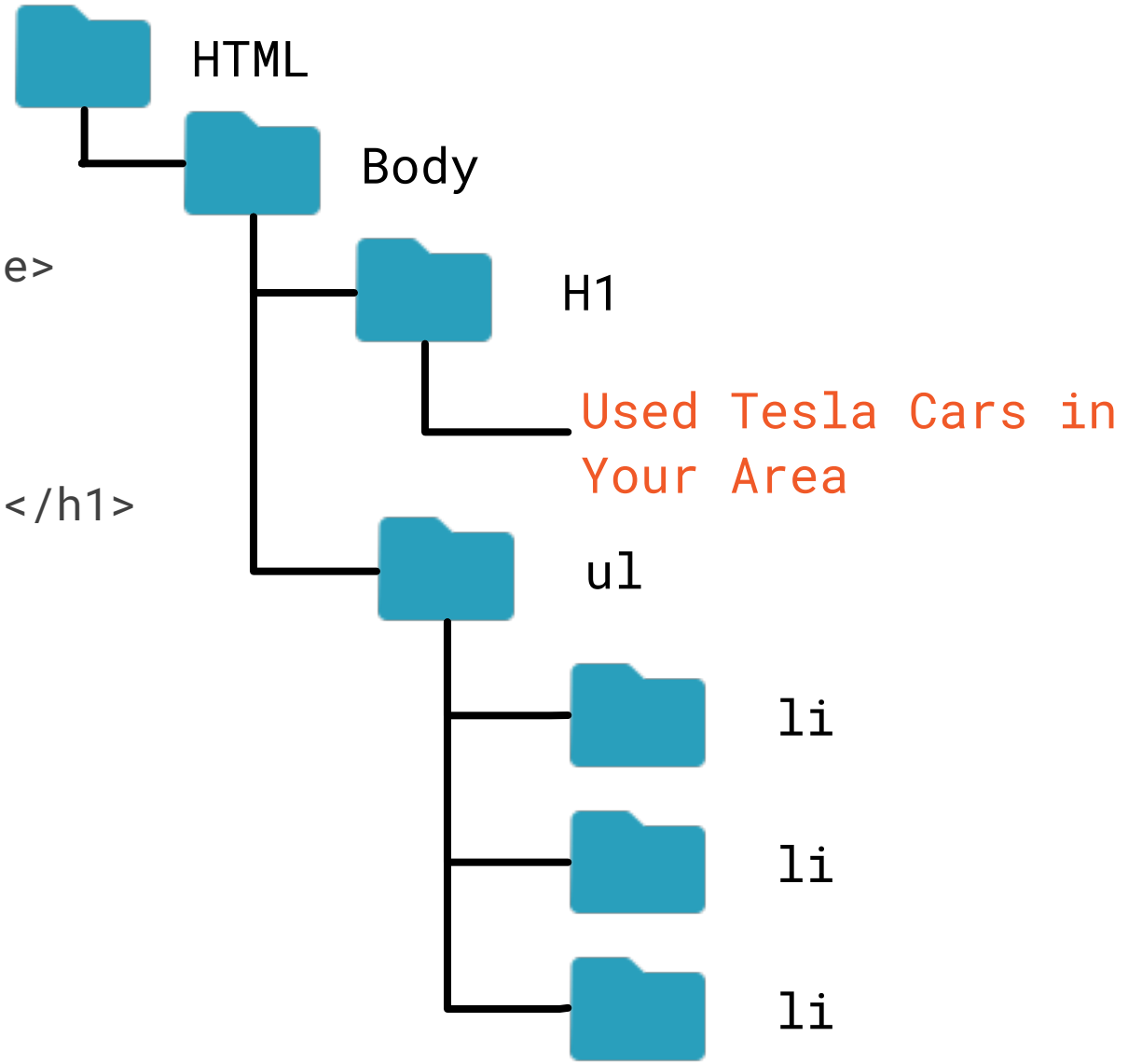
XPath Document Tree

```
<!DOCTYPE html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1>Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li> ... </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
</body>
</html>
```



XPath Document Tree

```
<!DOCTYPE html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1>Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li> ... </li>
        <li> ... </li>
        <li> ... </li>
    </ul>
</body>
</html>
```



XPath: Select Multiple

```
<!DOCTYPE html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1>Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <h2 class="model">2015 Tesla Model S</h2>
            <p class="description"><span class="price">$34,586</span>
                A wonderful car and a great deal.</p>
        </li>
        <li> ... </li>
```

```
css => 'li'
xpath => '//li'
```

XPath: Combined Selectors

```
<!DOCTYPE html>
<head>
    <title>Car Buying Web Page</title>
</head>

<body>
    <h1>Used Tesla Cars in Your Area</h1>
    <ul class="listings">
        <li id="vin3827" class="auto-listing">
            <h2 class="model">2015 Tesla Model S</h2>
            <p class="description"><span class="price">$34,586</span>
                A wonderful car and a great deal.</p>
        </li>
        <li> ... </li>
```

```
css => 'ul.listings li#vin3827'

xpath => '//ul[@class="listings"]/li[@id="vin3827"]'
```

XPath or CSS Selector

Which is Better?

Summary

HTML & CSS selectors

XPath selectors