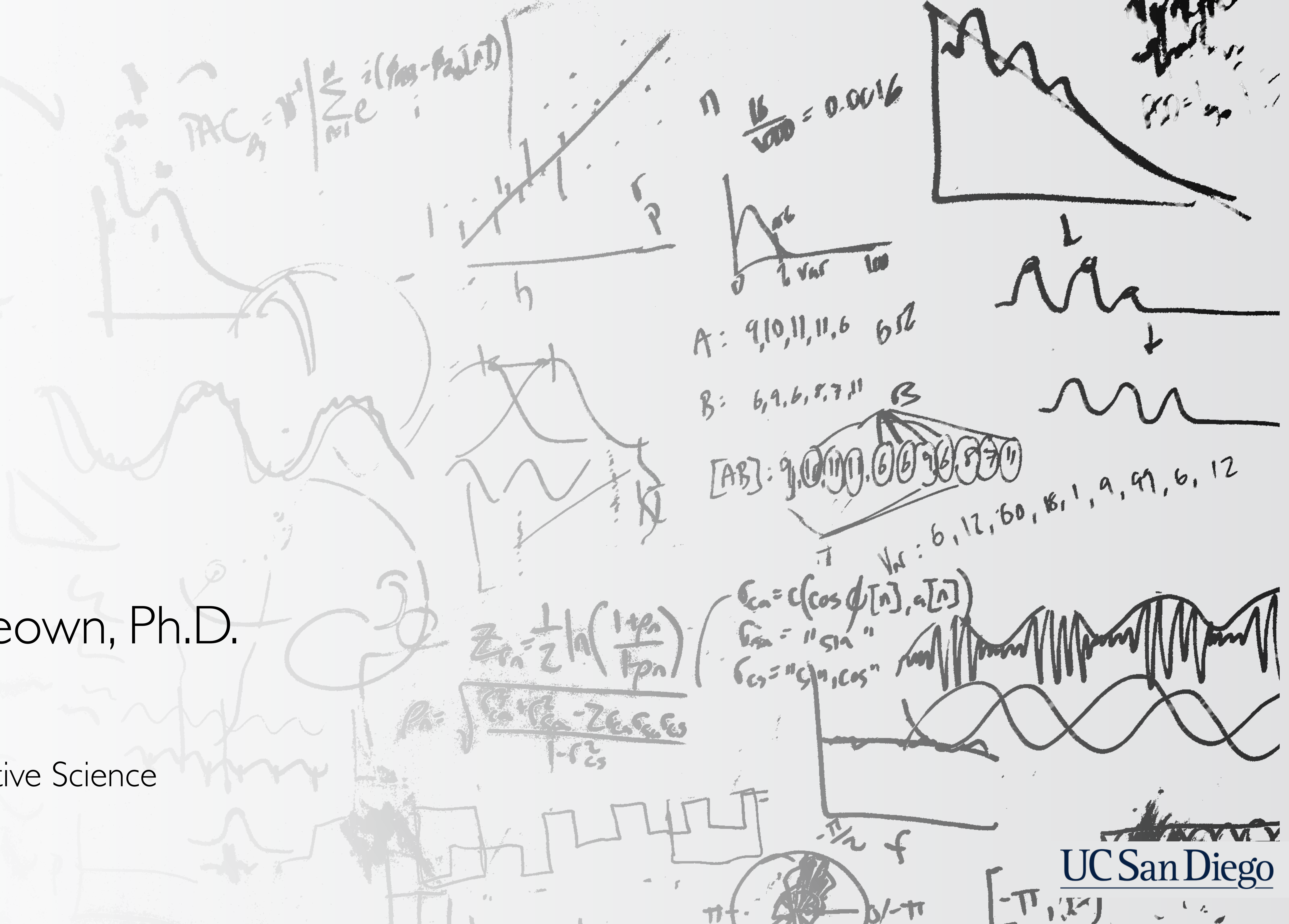


Christopher Keown, Ph.D.

UC San Diego

Department of Cognitive Science

ckeown@ucsd.edu

White Christmas

Weather forecasting is perhaps the most familiar domain of predictive modeling. Short-term forecasts are generally accurate, but what about longer-term prediction? What places will wake up to a snowy Christmas this year? And can you tell one month in advance?

Publicly available datasets

- *https://github.com/awesomedata/awesome-public-datasets*
- *https://medium.com/datadriveninvestor/the-50-best-public-datasets-for-machine-learning-d80e9f030279*
- *https://aws.amazon.com/opendata/public-datasets/*
- *https://registry.opendata.aws/*
- *https://www.data.gov/*

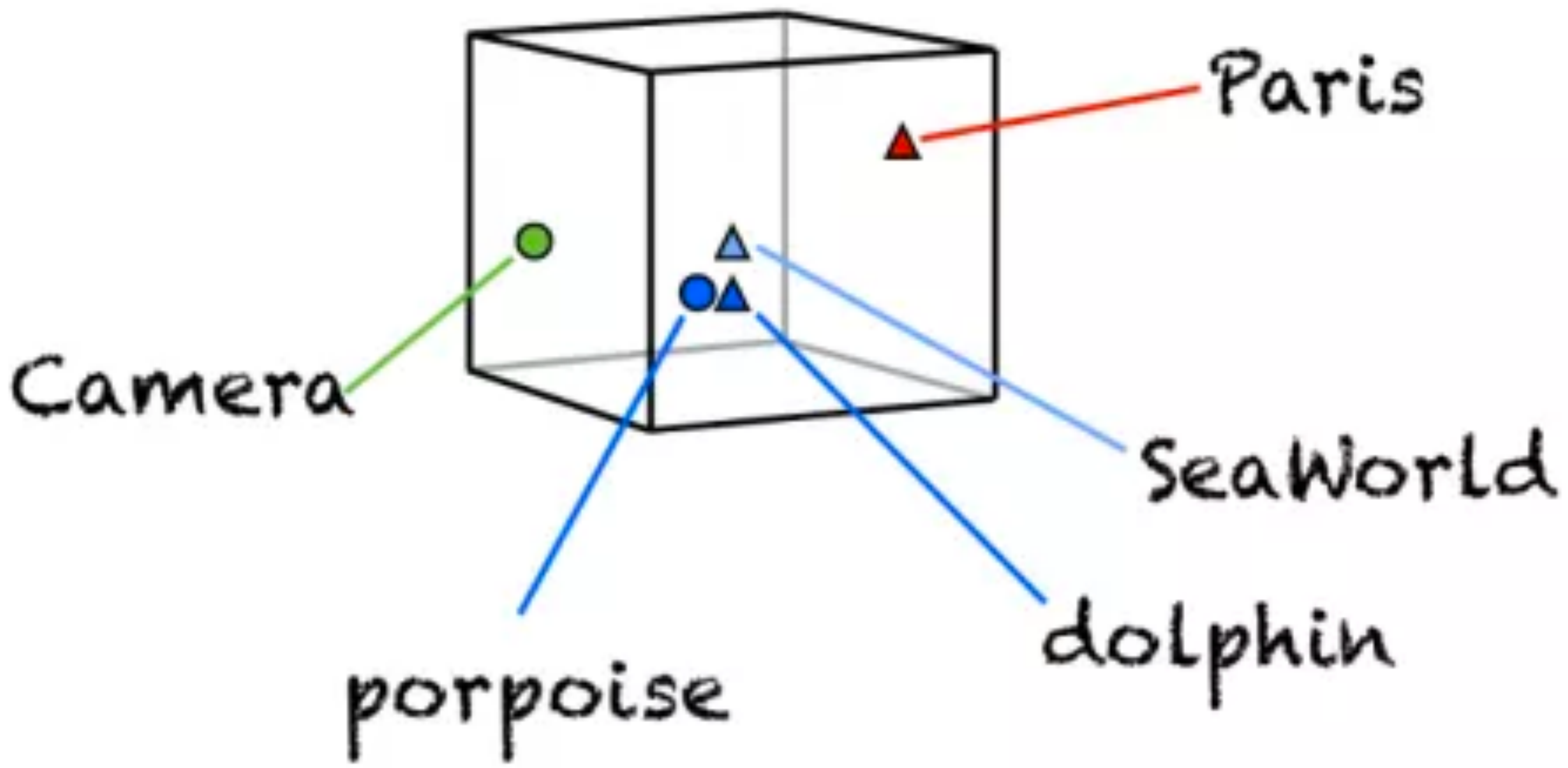
Types of data - quantitive vs. categorical

- ***Quantitative data*** *consists of numerical values, like height and weight.*
- ***Categorical data*** *consists of labels describing the properties of the objects under investigation, like gender, hair color, and occupation*
    - *Categorical data doesn’t have an order to it*
    - *Does it make any sense to talk about the maximum or minimum hair color? What is the interpretation of my hair color minus your hair color?*

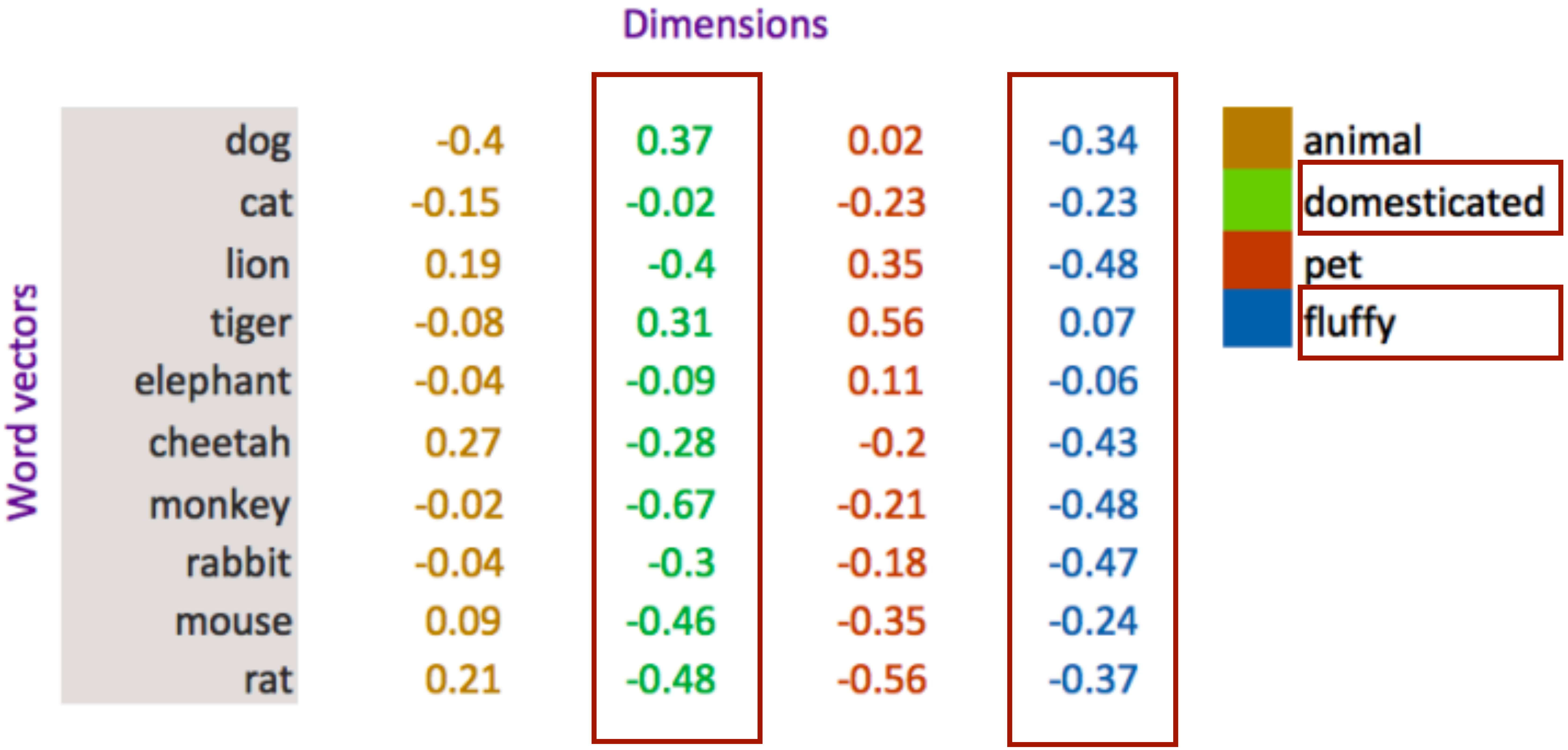
Quantitive or Categorical?

- *Favorite ice cream flavor*
- *Money spent in marketing budget*
- *GPA*
- *Letter grade*
- *Images*
- *Words*

Can we turn word into numbers?


Paris
Camera
SeaWorld
dolphin
porpoise

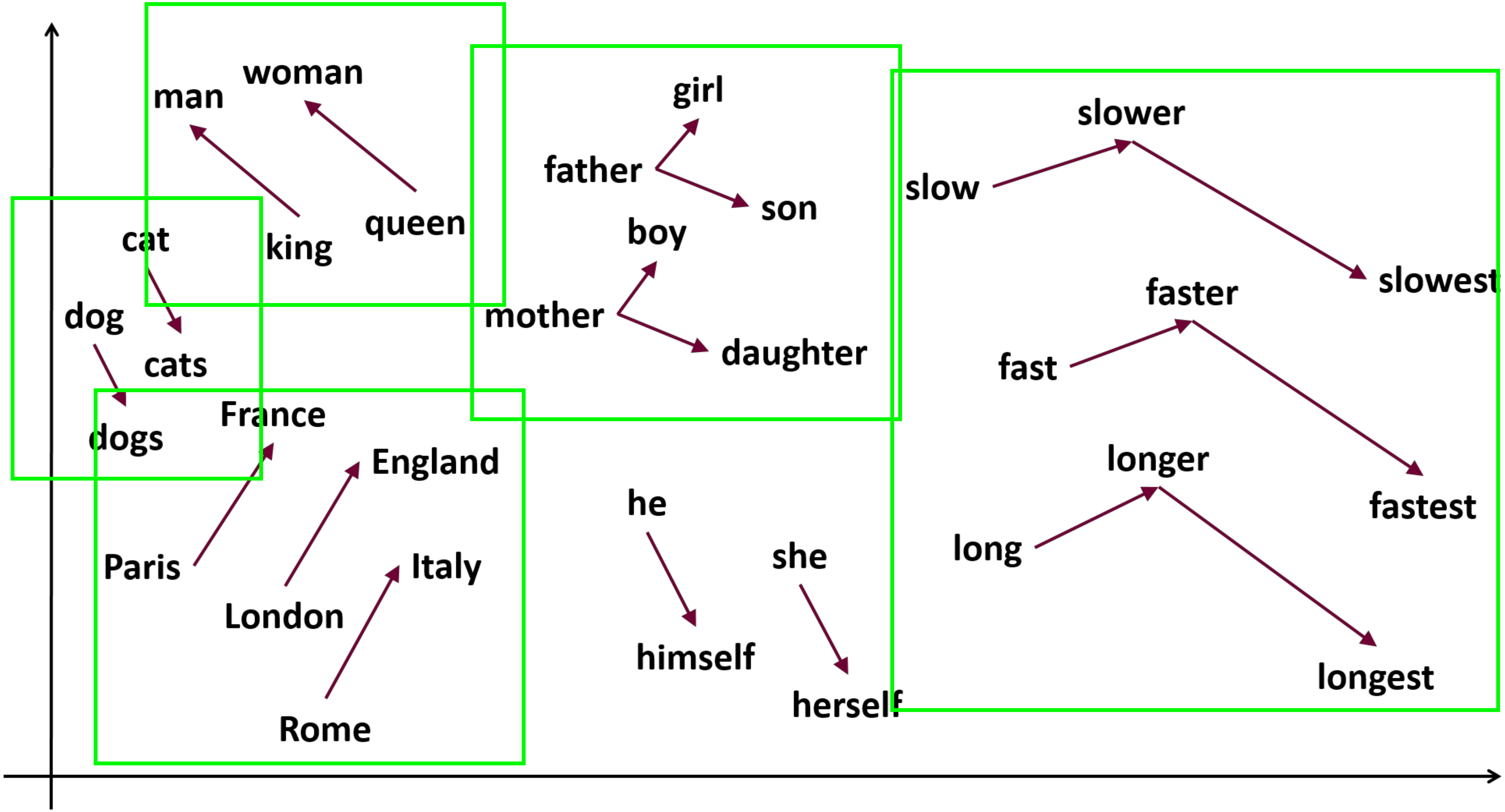
How can we represents words of animals as vectors?



Word vectors	Dimensions			
dog	-0.4	0.37	0.02	-0.34
cat	-0.15	-0.02	-0.23	-0.23
lion	0.19	-0.4	0.35	-0.48
tiger	-0.08	0.31	0.56	0.07
elephant	-0.04	-0.09	0.11	-0.06
cheetah	0.27	-0.28	-0.2	-0.43
monkey	-0.02	-0.67	-0.21	-0.48
rabbit	-0.04	-0.3	-0.18	-0.47
mouse	0.09	-0.46	-0.35	-0.24
rat	0.21	-0.48	-0.56	-0.37

- 🟫 animal
- 🟩 domesticated
- 🟥 pet
- 🟦 fluffy

Numerical properties preserved



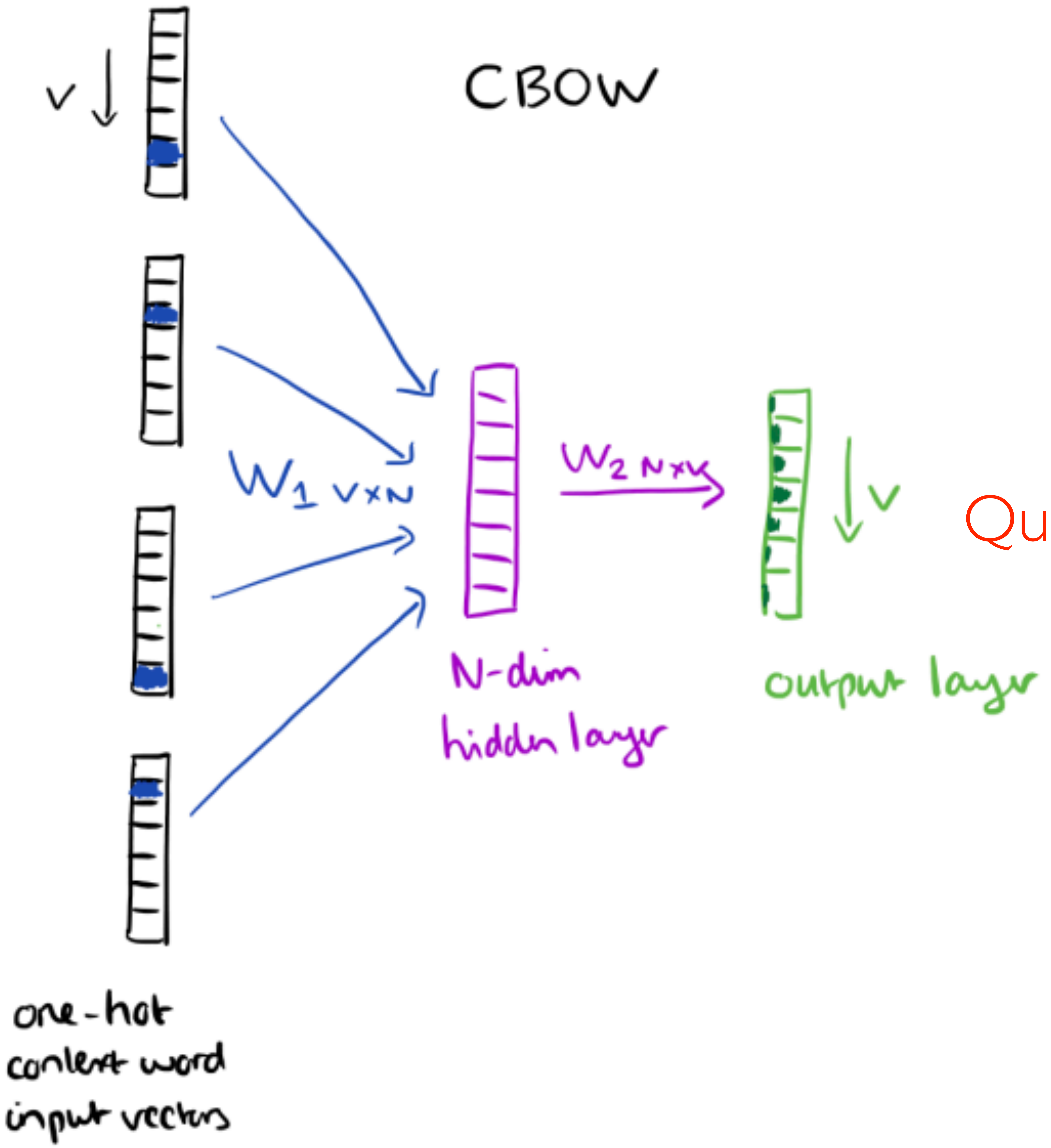
Word embedding - automate the process!

Word embedding is the collective name for a set of language modeling and feature learning techniques in natural language processing (NLP) where words or phrases from the vocabulary are mapped to vectors of real numbers (Wikipedia).

Categorical

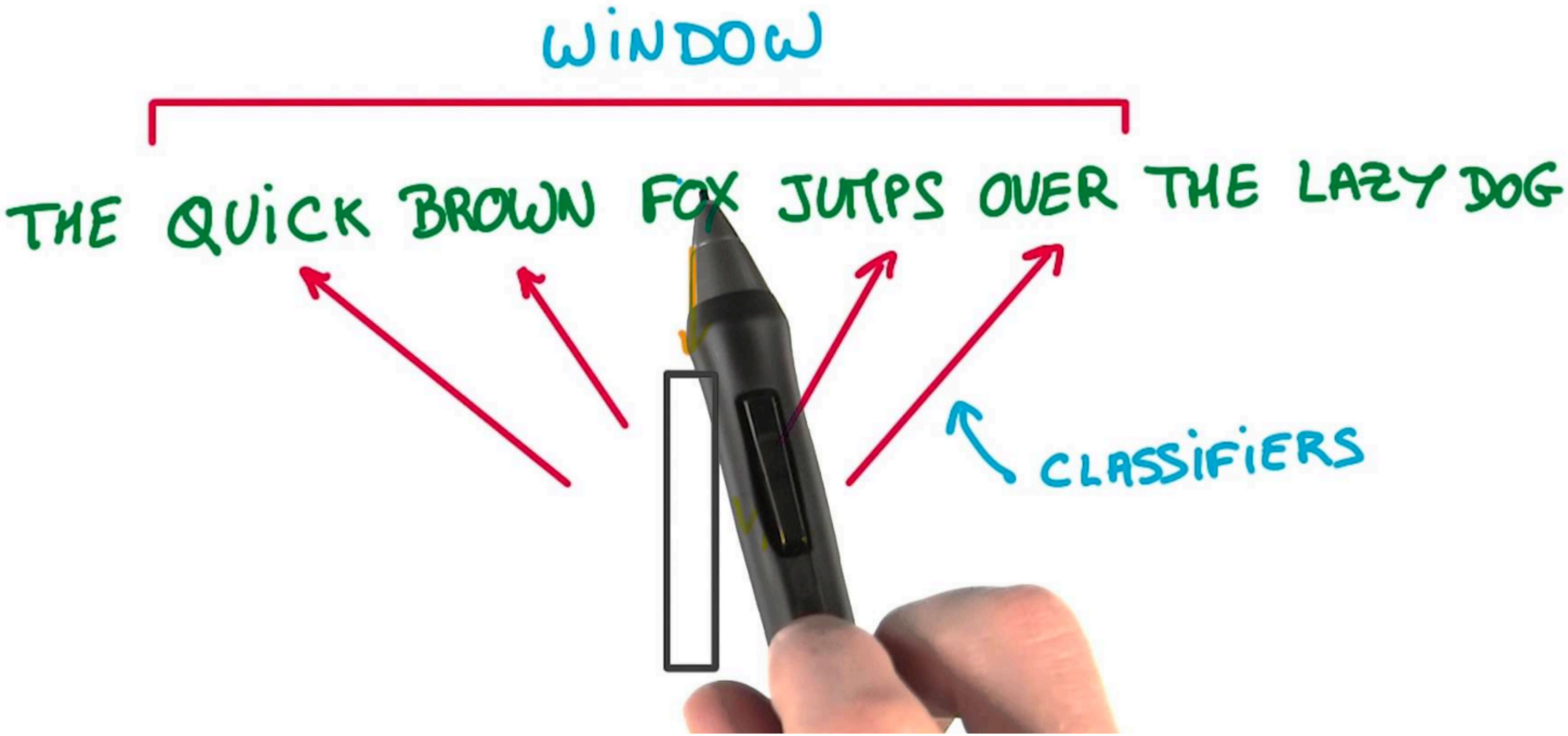
One-hot encoding

	1	2	3	4	5	6	7	8	9
man	1	0	0	0	0	0	0	0	0
woman	0	1	0	0	0	0	0	0	0
boy	0	0	1	0	0	0	0	0	0
girl	0	0	0	1	0	0	0	0	0
prince	0	0	0	0	1	0	0	0	0
princess	0	0	0	0	0	1	0	0	0
queen	0	0	0	0	0	0	1	0	0
king	0	0	0	0	0	0	0	1	0
monarch	0	0	0	0	0	0	0	0	1

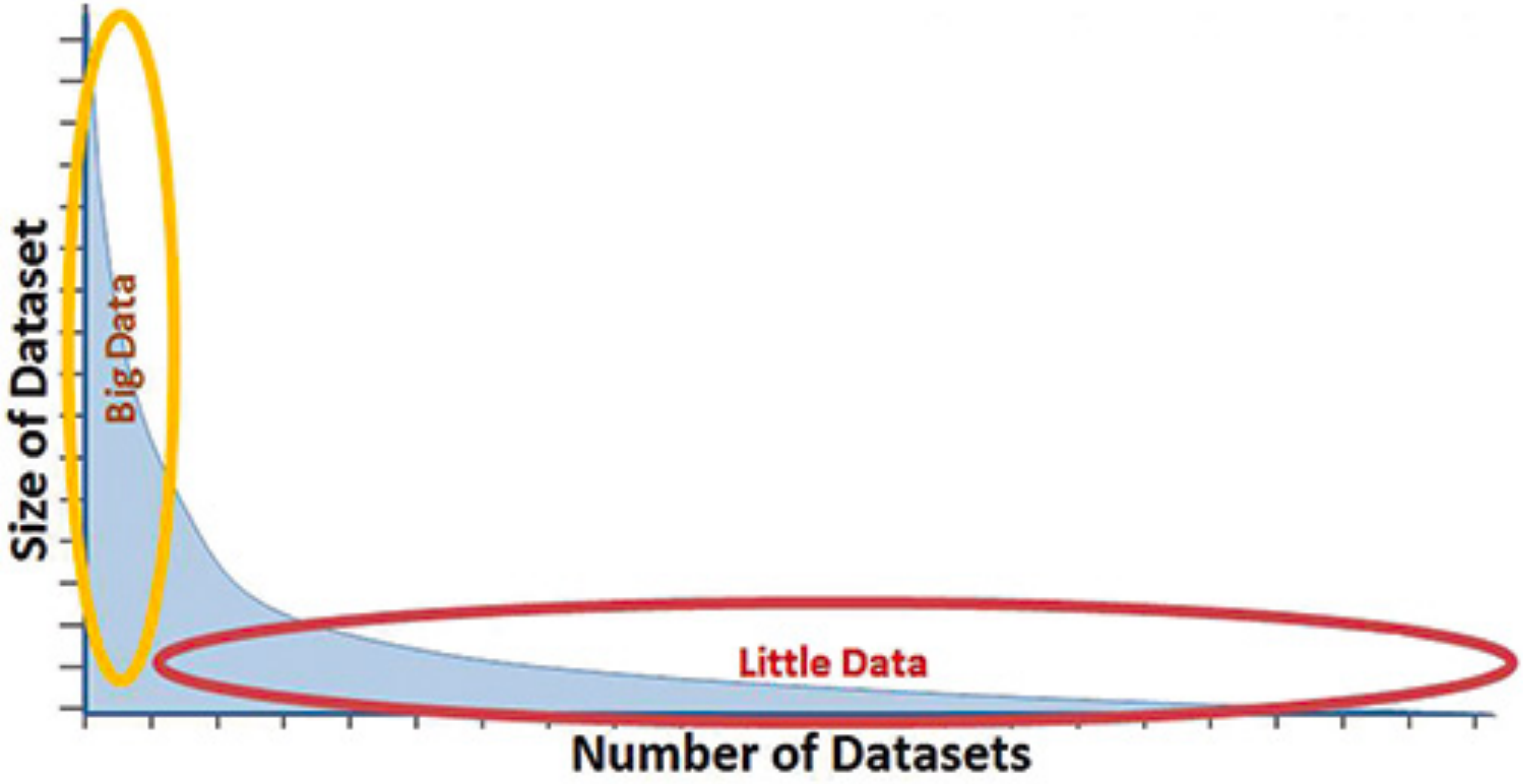


Quantitative

WORD2VEC


WINDOW
THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG
CLASSIFIERS

Types of data - big vs. little



Types of data - big vs. little

- *There are difficulties in working with large data sets.*
  - *The analysis cycle time slows as data size grows (slow to iterate)*
  - *Large data sets are complex to visualize*
- *Simple models do not require massive data to fit or evaluate*

The big vs. little data approach

What are voter preferences about the demographic presidential campaign pool?

The big vs. little data approach

Which approach do you think will be more accurate?

Take away: The right data set is the one most directly relevant to the tasks at hand, not necessarily the biggest one.

Types of questions - classification

Classification: *Often we seek to assign a label to an item from a discrete set of possibilities. Such problems as predicting the winner of a particular sporting contest (team A or team B?) or deciding the genre of a given movie (comedy, drama, or animation?) are classification problems, since each entail selecting a label from the possible choices.*

Types of questions - regression

Regression: *Another common task is to forecast a given numerical quantity. Predicting a person's weight or how much snow we will get this year is a regression problem, where we forecast the future value of a numerical function in terms of previous values and other relevant features.*

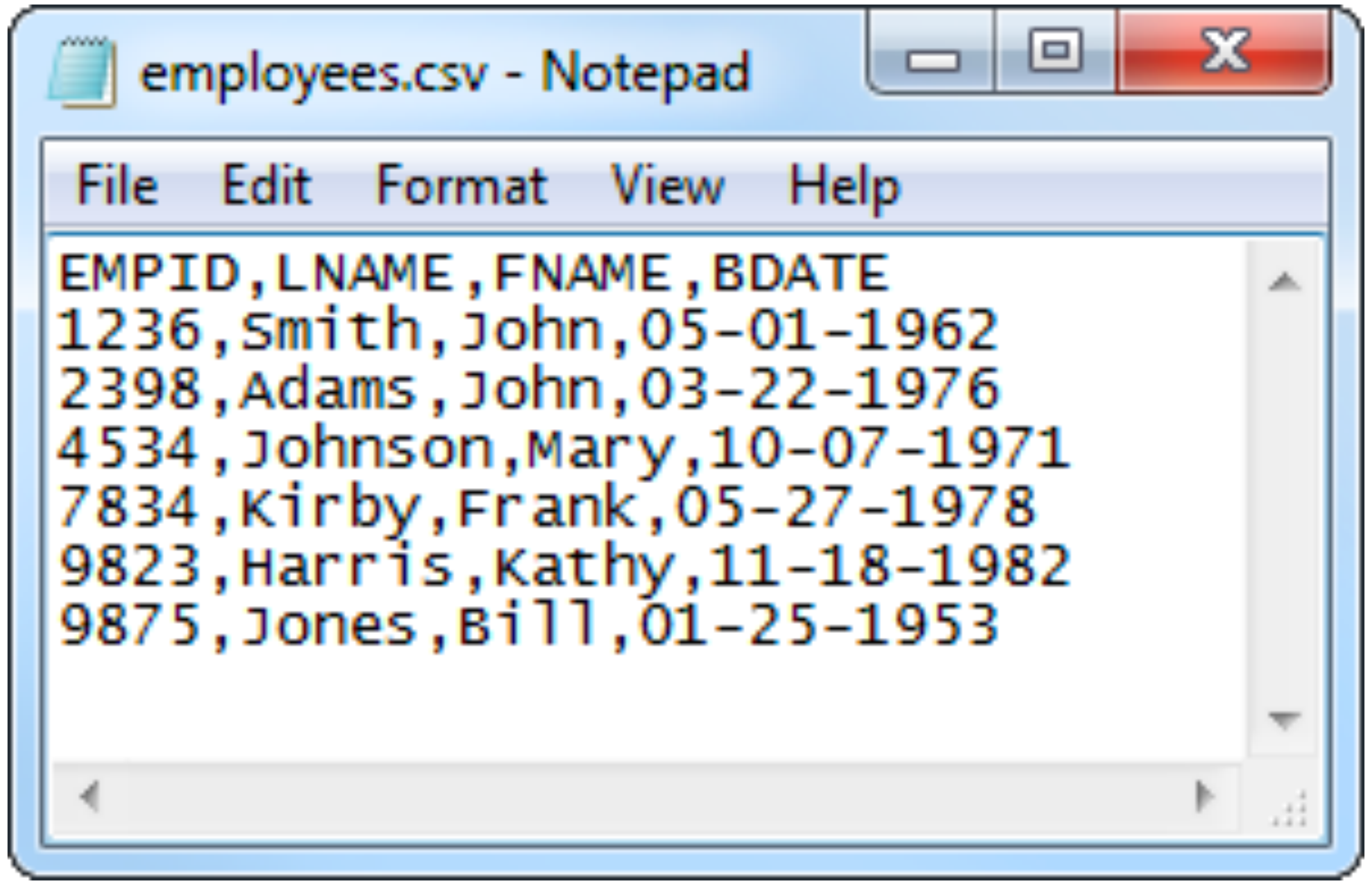
Regression or classification?

- *Will the price of a particular stock be higher or lower tomorrow?*
- *What will the price of a particular stock be tomorrow?*
- *Is this person a good risk to sell an insurance policy to?*
- *How long do we expect this person to live?*

Types of data - structured vs. unstructured

- ***Structured data*** *- data sets that are structured, like the tables in a database or spread- sheet program.*

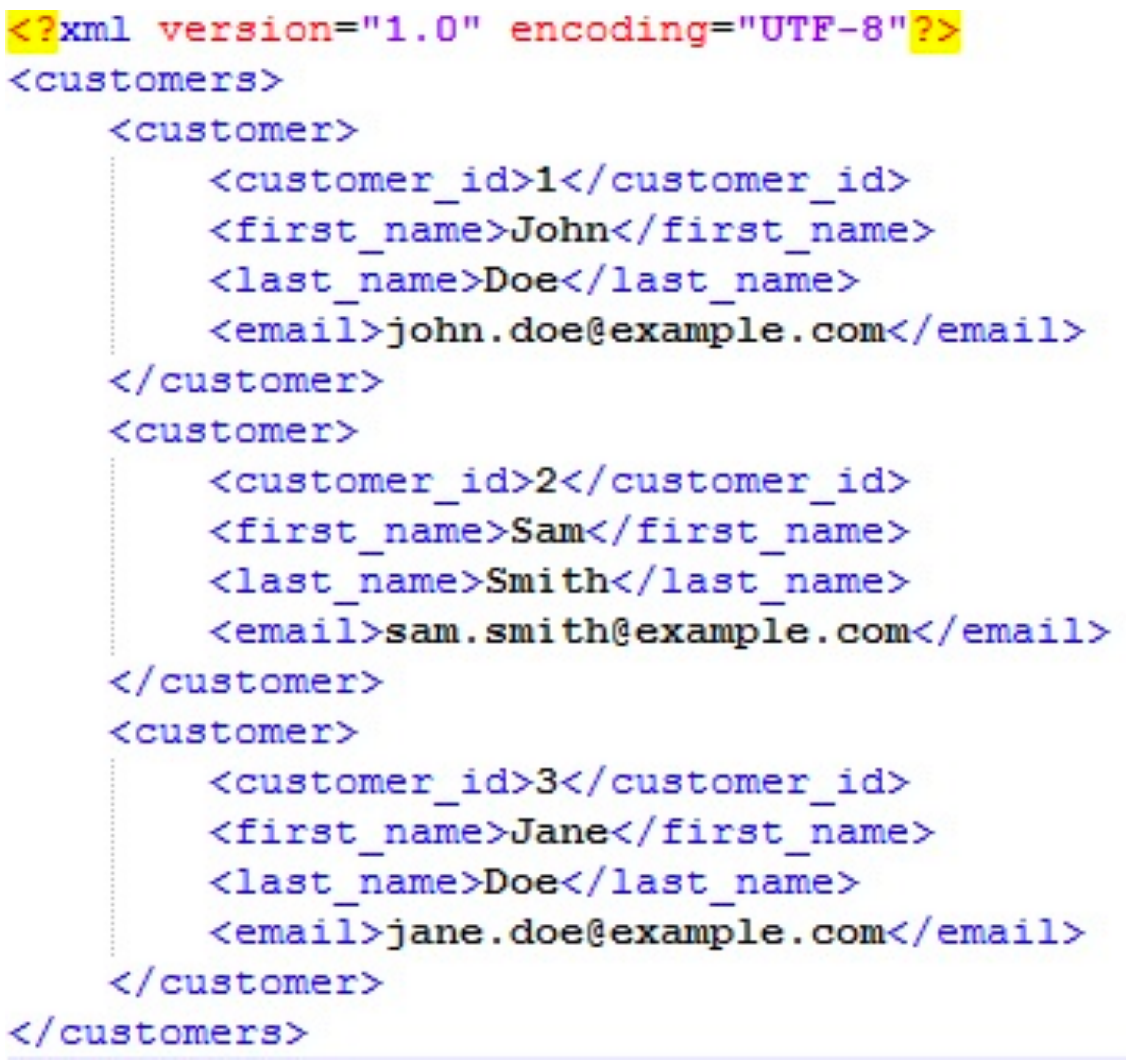
CSV and TSV



```
EMPID,LNAME,FNAME,BDATE
1236,Smith,John,05-01-1962
2398,Adams,John,03-22-1976
4534,Johnson,Mary,10-07-1971
7834,Kirby,Frank,05-27-1978
9823,Harris,Kathy,11-18-1982
9875,Jones,Bill,01-25-1953
```

Lightweight but not good for hierarchical data

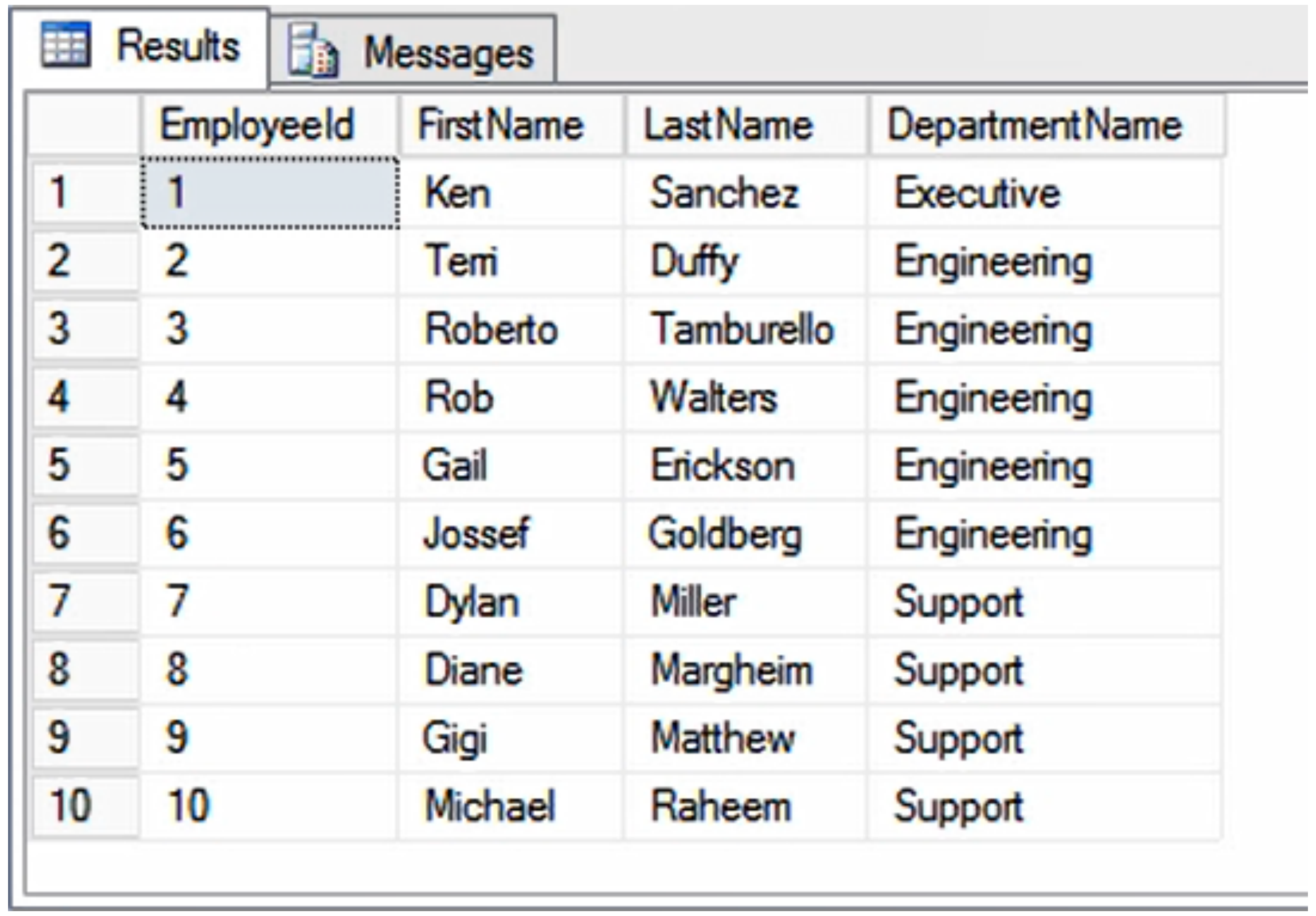
XML-formatted data



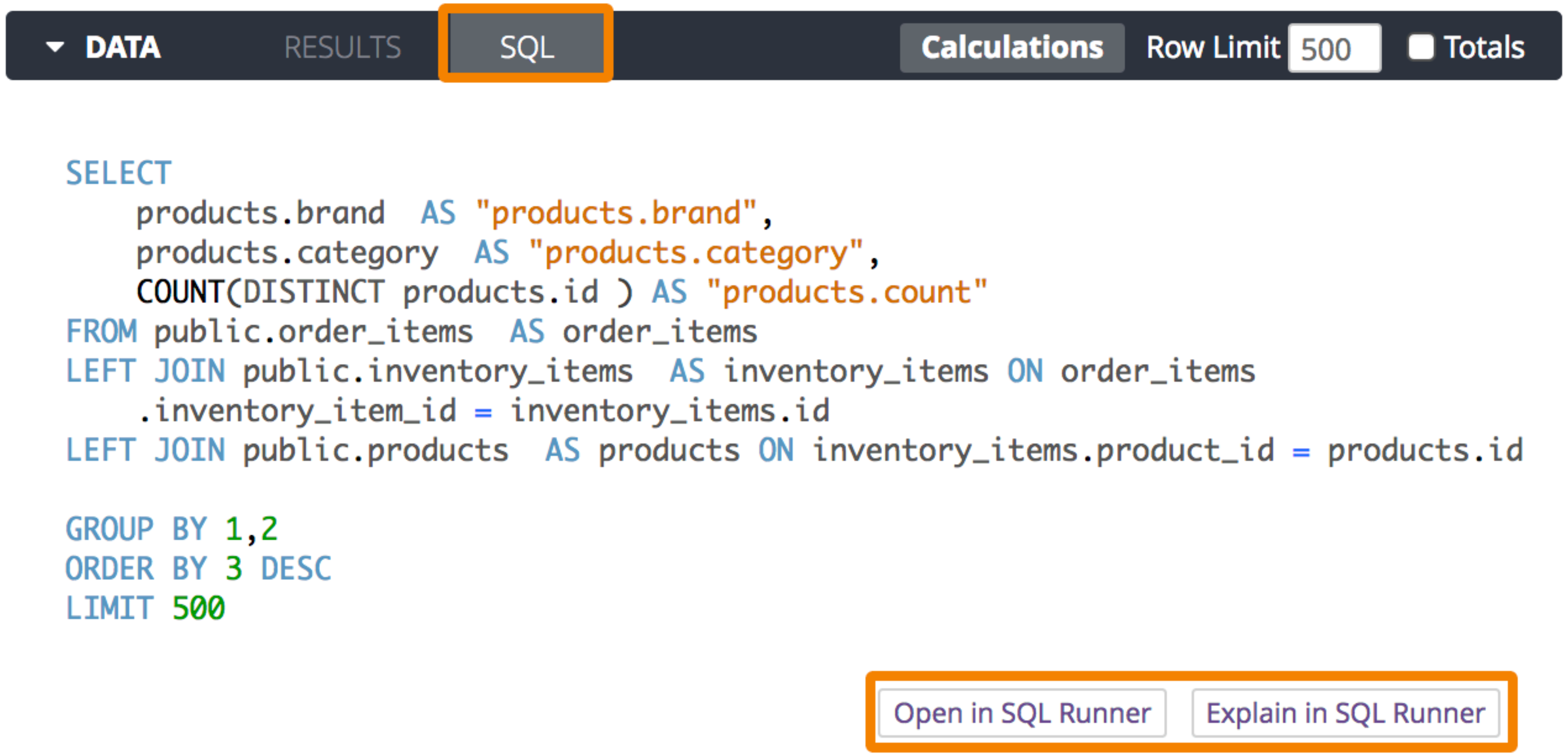
```
<?xml version="1.0" encoding="UTF-8"?>
<customers>
    <customer>
        <customer_id>1</customer_id>
        <first_name>John</first_name>
        <last_name>Doe</last_name>
        <email>john.doe@example.com</email>
    </customer>
    <customer>
        <customer_id>2</customer_id>
        <first_name>Sam</first_name>
        <last_name>Smith</last_name>
        <email>sam.smith@example.com</email>
    </customer>
    <customer>
        <customer_id>3</customer_id>
        <first_name>Jane</first_name>
        <last_name>Doe</last_name>
        <email>jane.doe@example.com</email>
    </customer>
</customers>
```

Not good for big data

SQL - Structured Query Language



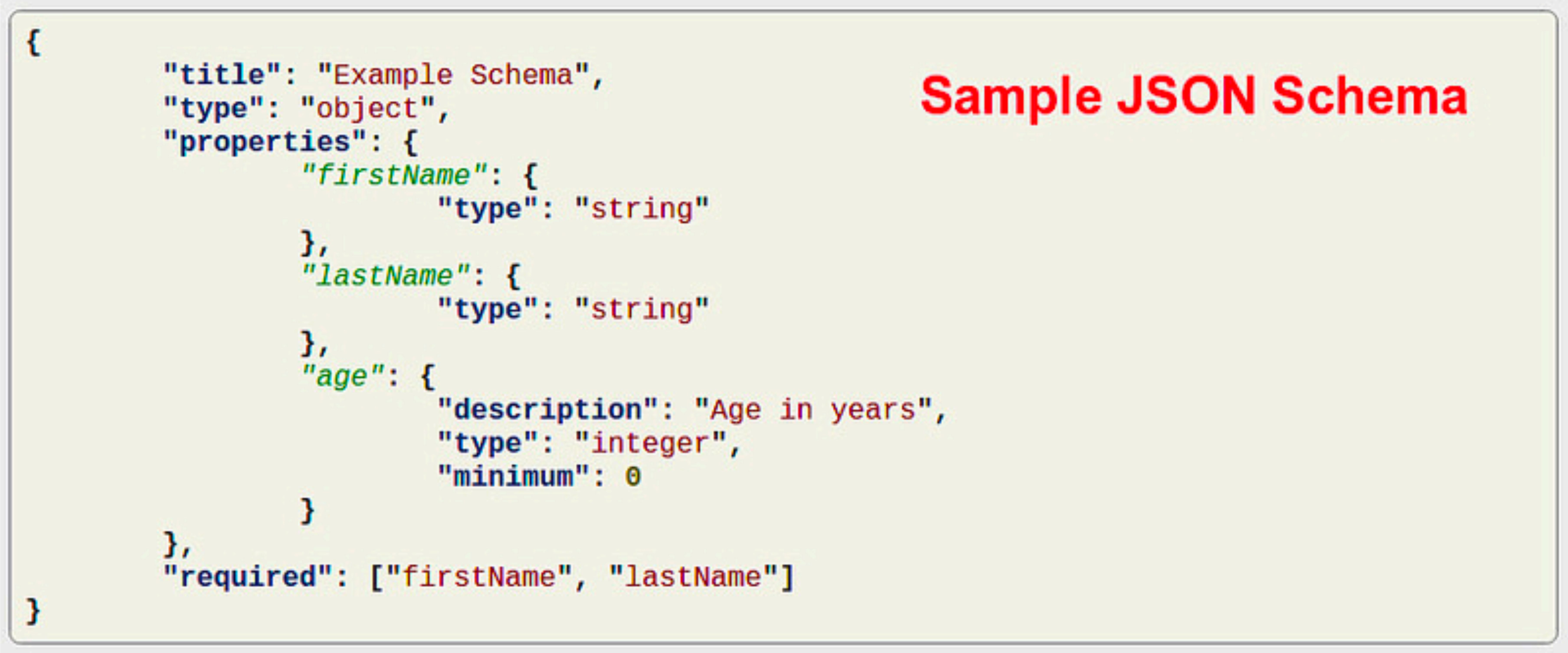
	EmployeeId	FirstName	LastName	DepartmentName
1	1	Ken	Sanchez	Executive
2	2	Terri	Duffy	Engineering
3	3	Roberto	Tamburello	Engineering
4	4	Rob	Walters	Engineering
5	5	Gail	Erickson	Engineering
6	6	Jossef	Goldberg	Engineering
7	7	Dylan	Miller	Support
8	8	Diane	Margheim	Support
9	9	Gigi	Matthew	Support
10	10	Michael	Raheem	Support



```
SELECT
    products.brand  AS "products.brand",
    products.category  AS "products.category",
    COUNT(DISTINCT products.id ) AS "products.count"
FROM public.order_items  AS order_items
LEFT JOIN public.inventory_items  AS inventory_items ON order_items
    .inventory_item_id = inventory_items.id
LEFT JOIN public.products  AS products ON inventory_items.product_id = products.id

GROUP BY 1,2
ORDER BY 3 DESC
LIMIT 500
```

JSON

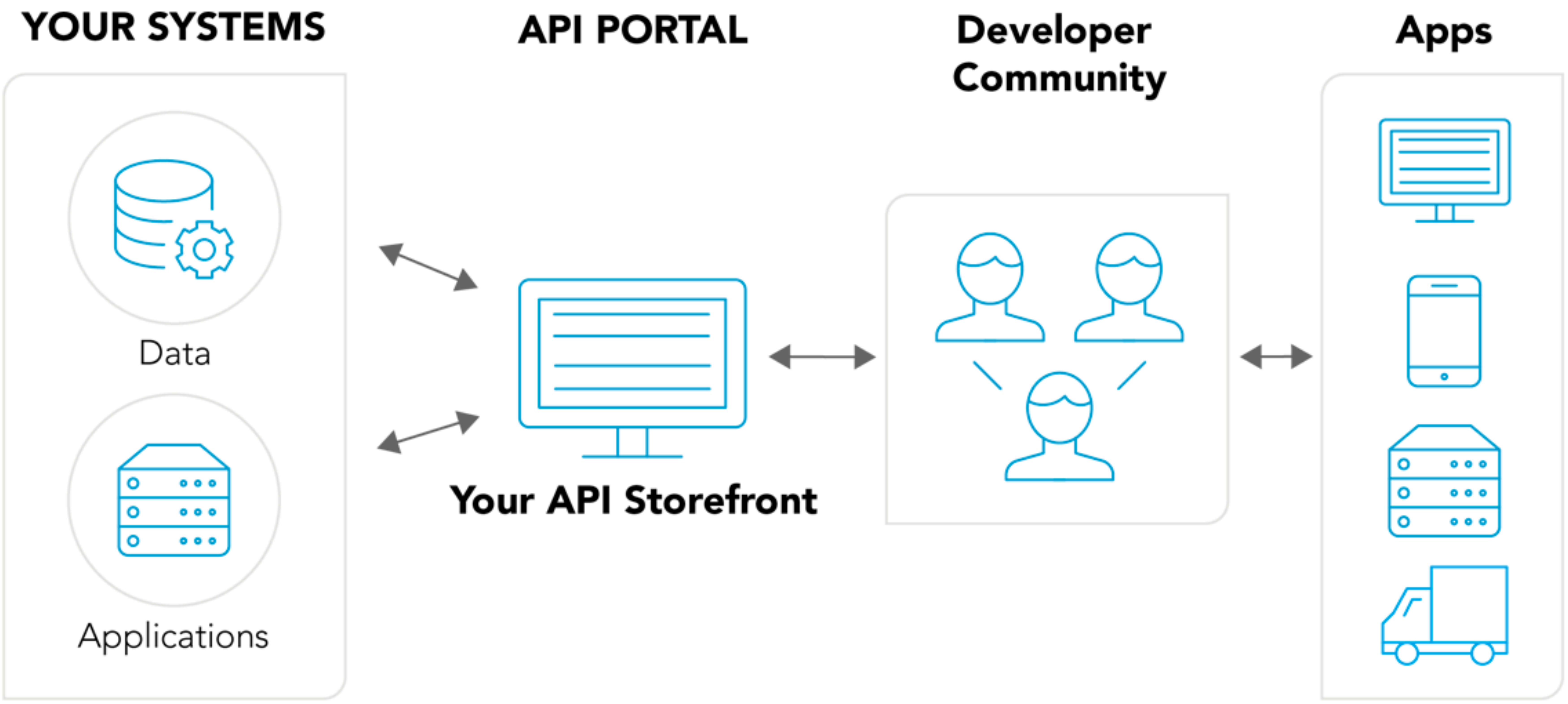


```
{
        "title": "Example Schema",
        "type": "object",
        "properties": {
                "firstName": {
                        "type": "string"
                },
                "lastName": {
                        "type": "string"
                },
                "age": {
                        "description": "Age in years",
                        "type": "integer",
                        "minimum": 0
                }
        },
        "required": ["firstName", "lastName"]
}
```

Sample JSON Schema

Good for hierarchical data

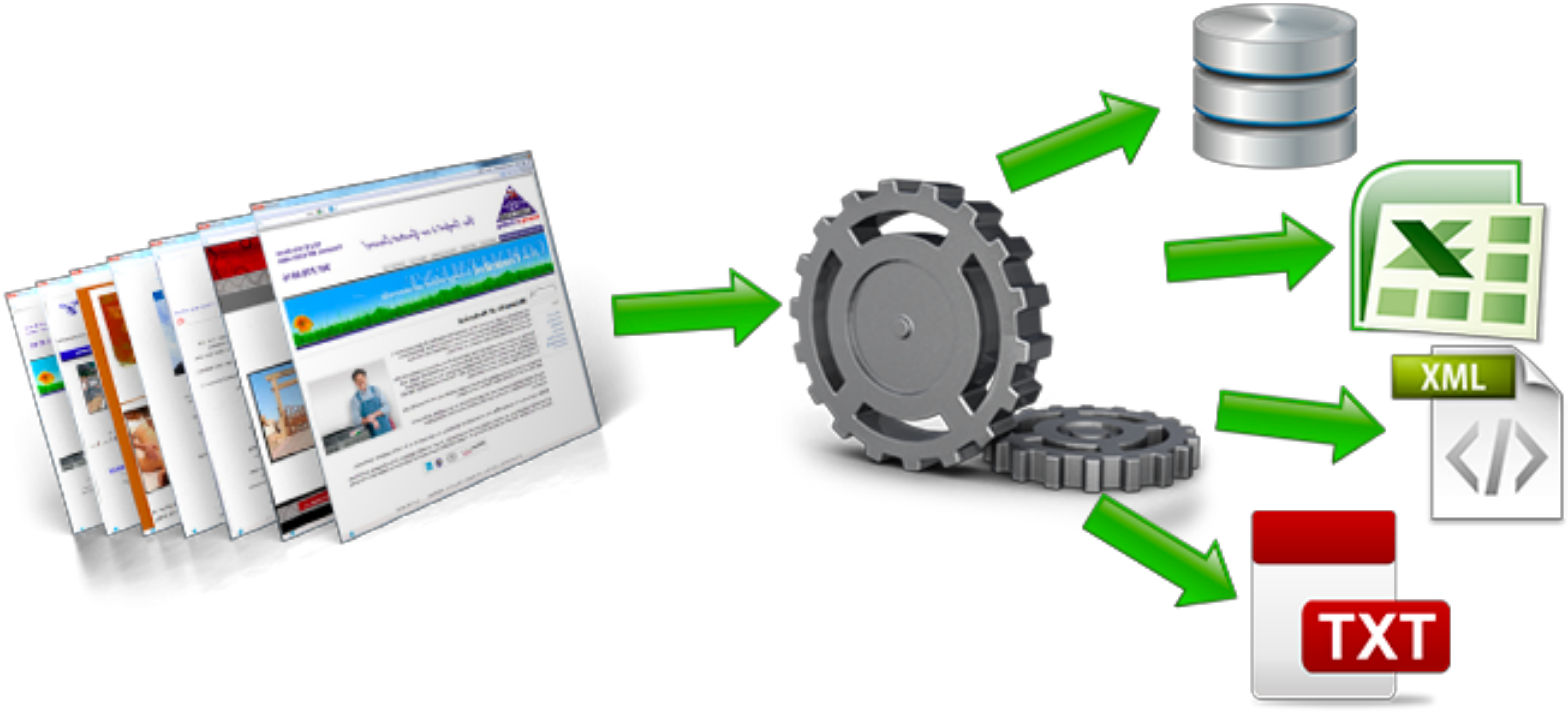
API - Application Programming Interface



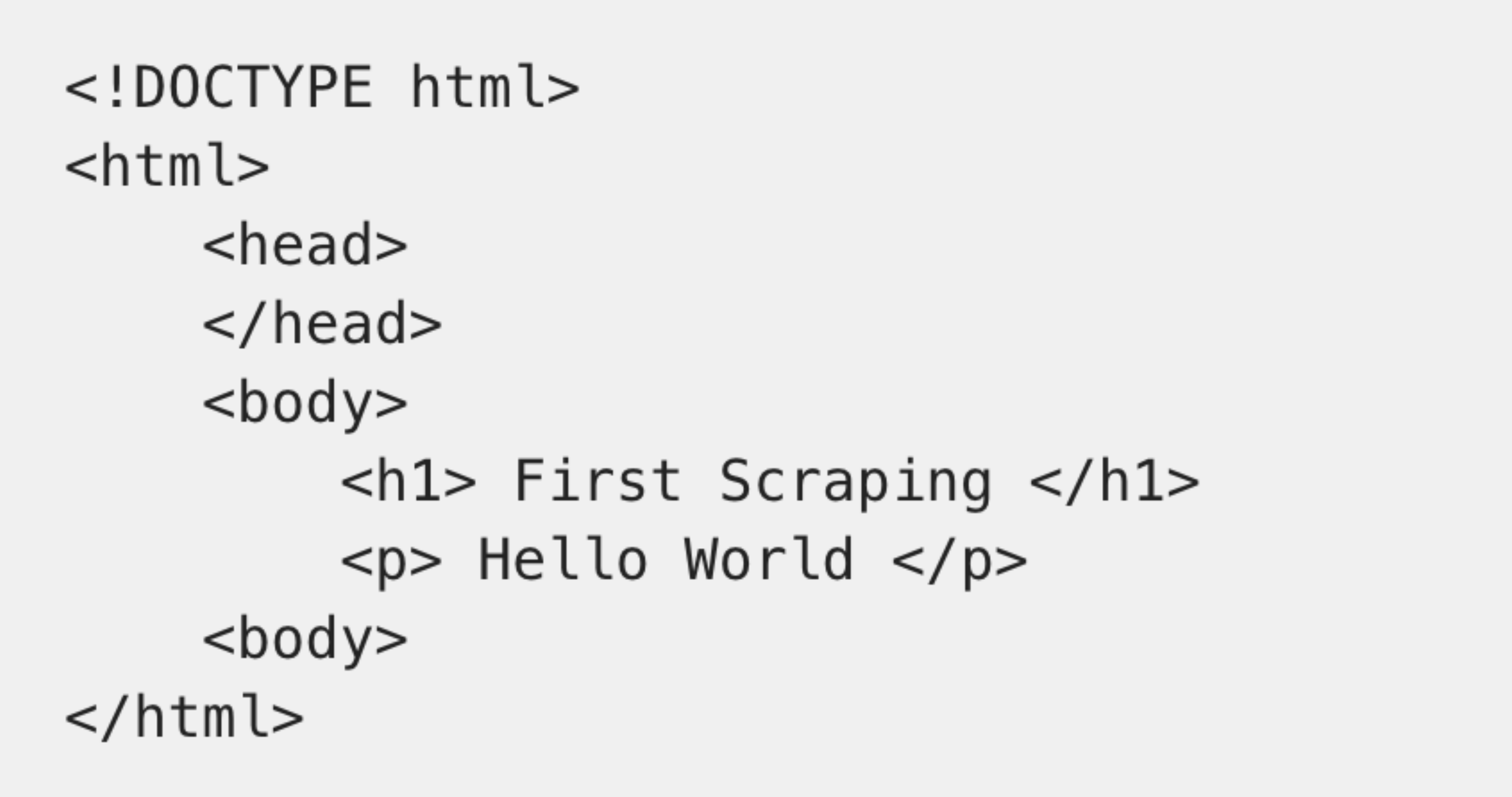
Types of data - structured vs. unstructured

Unstructured data *- Some datasets record information about the state of the world, but in a more heterogeneous way. Perhaps it is a large text corpus with images and links like Wikipedia, or the complicated mix of notes and test results appearing in personal medical records.*

Data Scraping

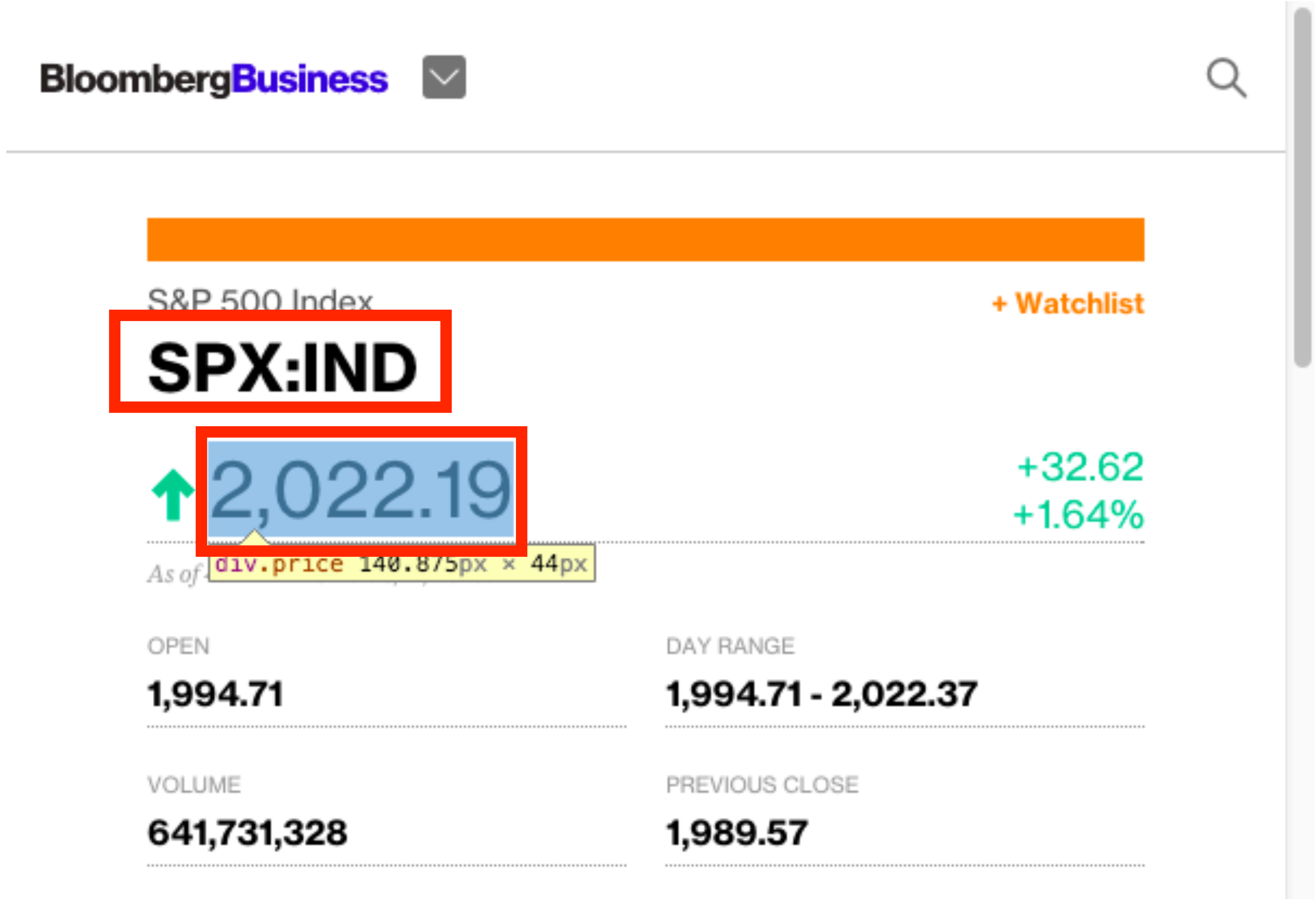


Data Scraping



```
<!DOCTYPE html>
<html>
    <head>
    </head>
    <body>
        <h1> First Scraping </h1>
        <p> Hello World </p>
    <body>
</html>
```

Getting stock names & prices from Bloomberg



S&P 500 Index

SPX:IND

2,022.19

+32.62
+1.64%

div.price 140.875px × 44px

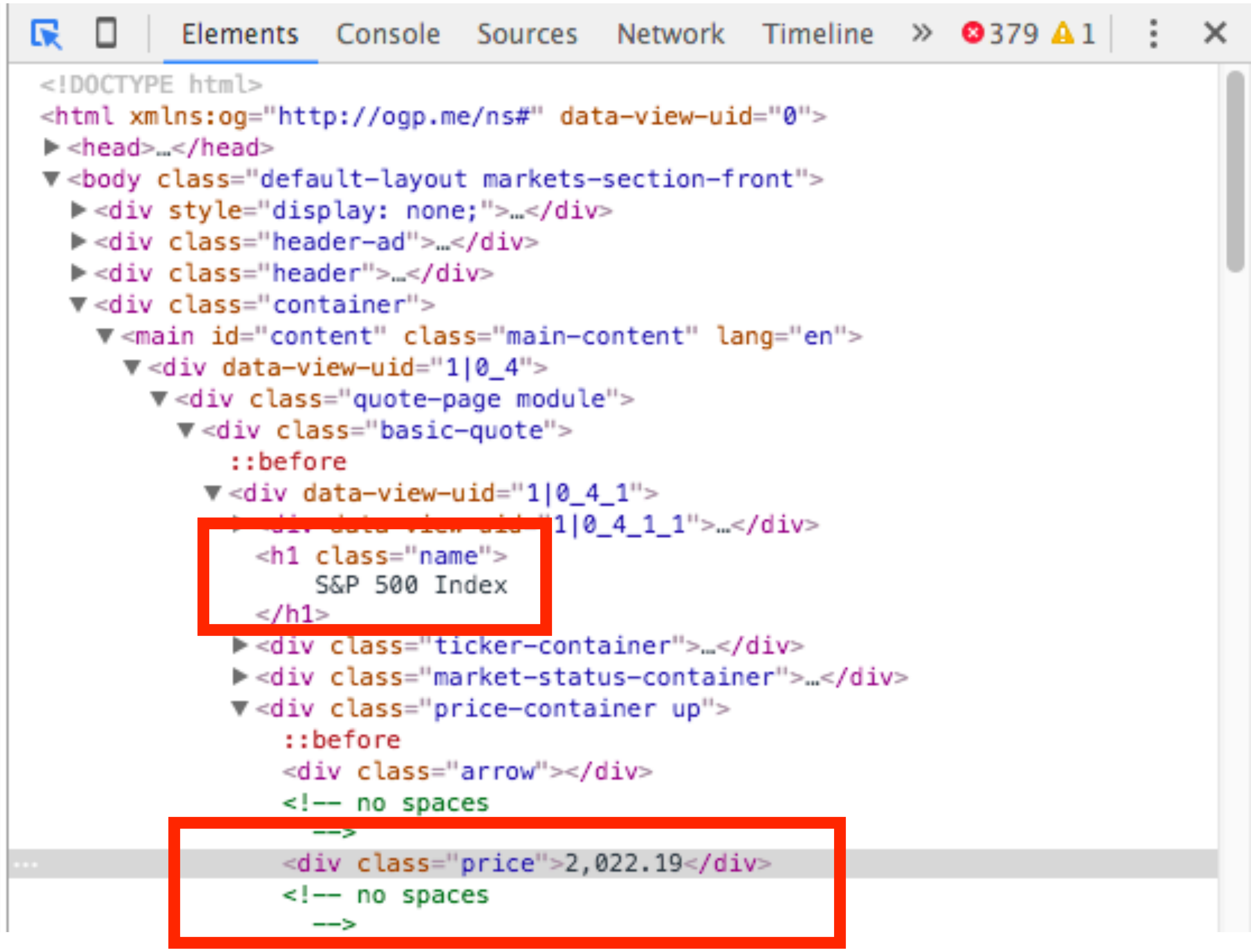
As of

OPEN
1,994.71

DAY RANGE
1,994.71 - 2,022.37

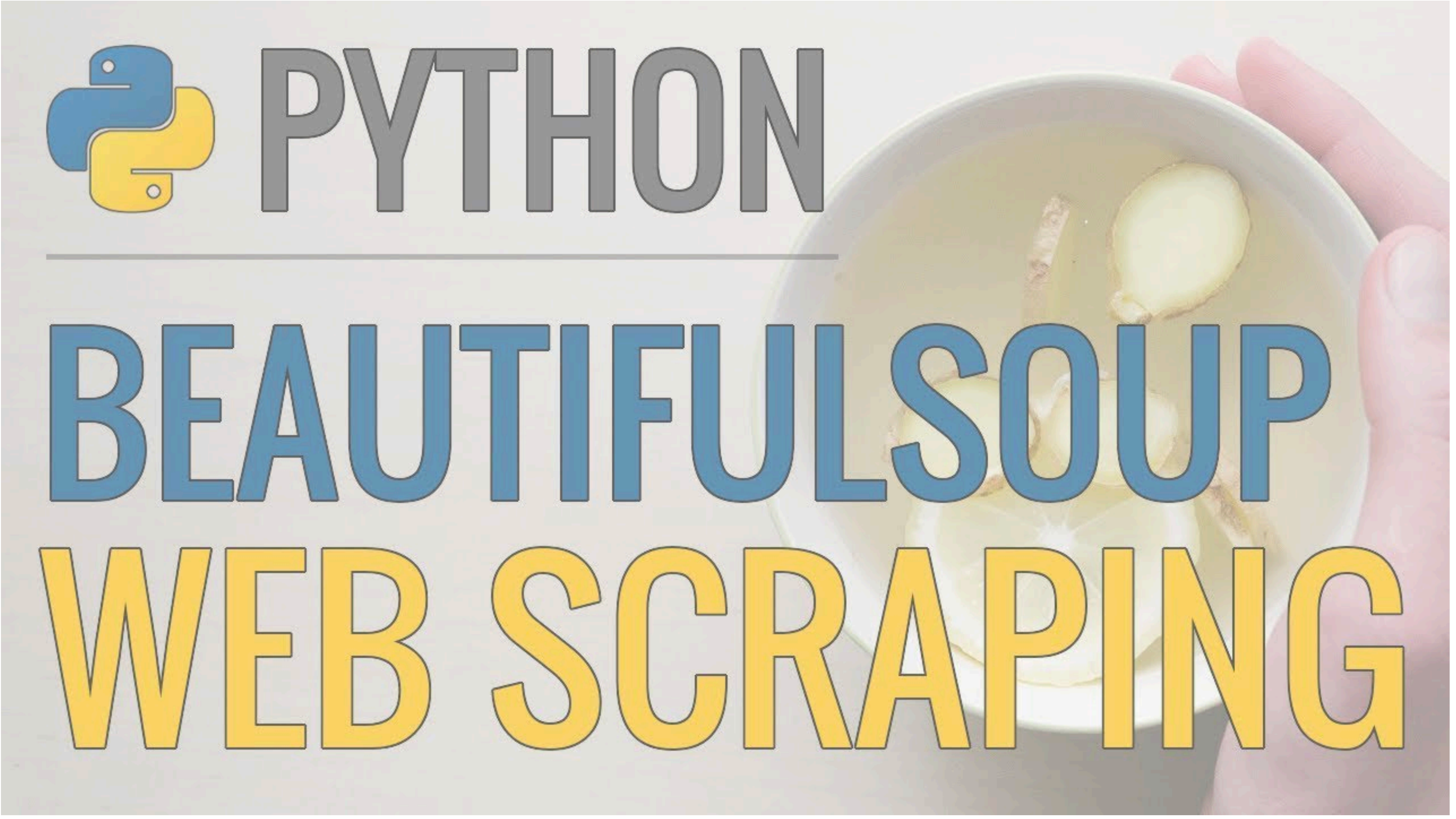
VOLUME
641,731,328

PREVIOUS CLOSE
1,989.57

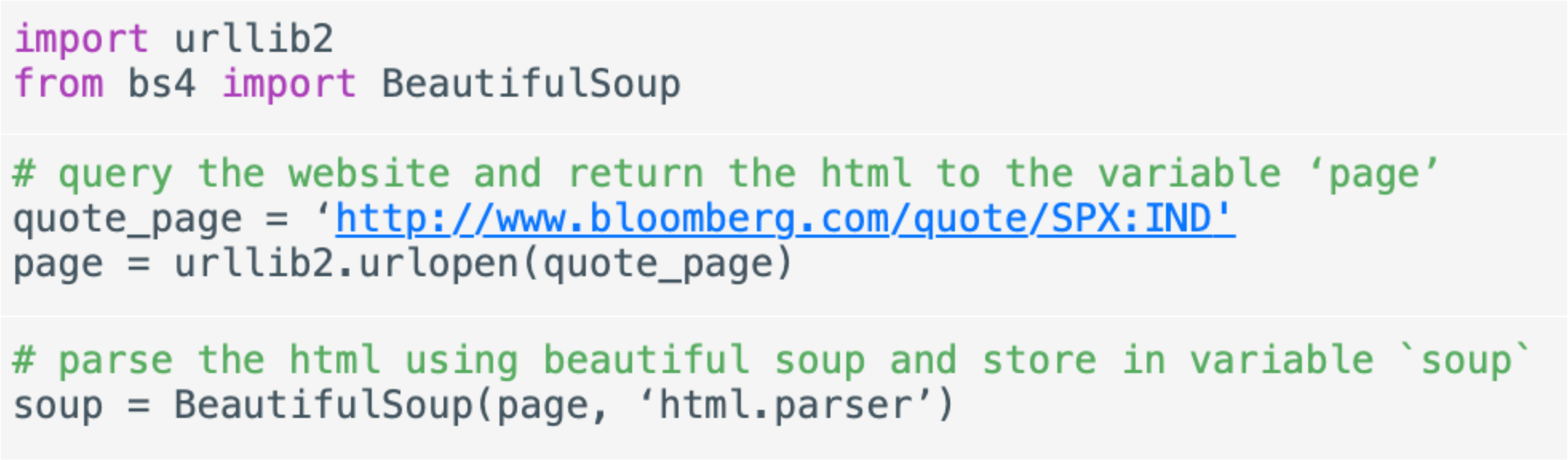


```
<!DOCTYPE html>
<html xmlns:og="http://ogp.me/ns#" data-view-uid="0">
▶<head>…</head>
▼<body class="default-layout markets-section-front">
  ▶<div style="display: none;">…</div>
  ▶<div class="header-ad">…</div>
  ▶<div class="header">…</div>
  ▼<div class="container">
    ▼<main id="content" class="main-content" lang="en">
      ▼<div data-view-uid="1|0_4">
        ▼<div class="quote-page module">
          ▼<div class="basic-quote">
              ::before
            ▼<div data-view-uid="1|0_4_1">
              ▶<div data-view-uid="1|0_4_1_1">…</div>
                <h1 class="name">
                    S&P 500 Index
                </h1>
              ▶<div class="ticker-container">…</div>
              ▶<div class="market-status-container">…</div>
              ▼<div class="price-container up">
                  ::before
                  <div class="arrow"></div>
                  <!-- no spaces
                    -->
                  <div class="price">2,022.19</div>
                  <!-- no spaces
                    -->
```

Data Scraping



Data Scraping

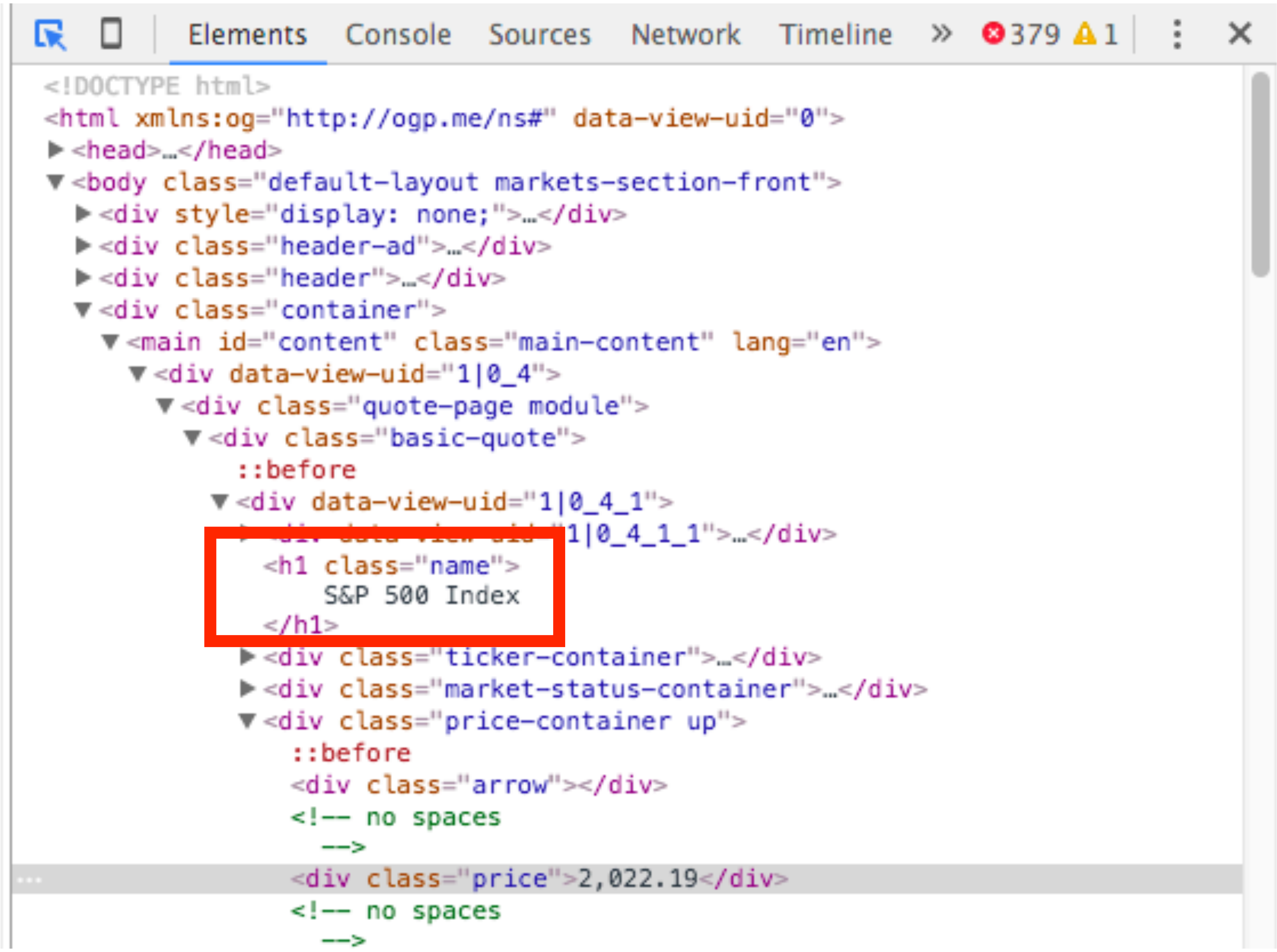


```
import urllib2
from bs4 import BeautifulSoup

# query the website and return the html to the variable ‘page’
quote_page = ‘http://www.bloomberg.com/quote/SPX:IND'
page = urllib2.urlopen(quote_page)

# parse the html using beautiful soup and store in variable `soup`
soup = BeautifulSoup(page, ‘html.parser’)
```

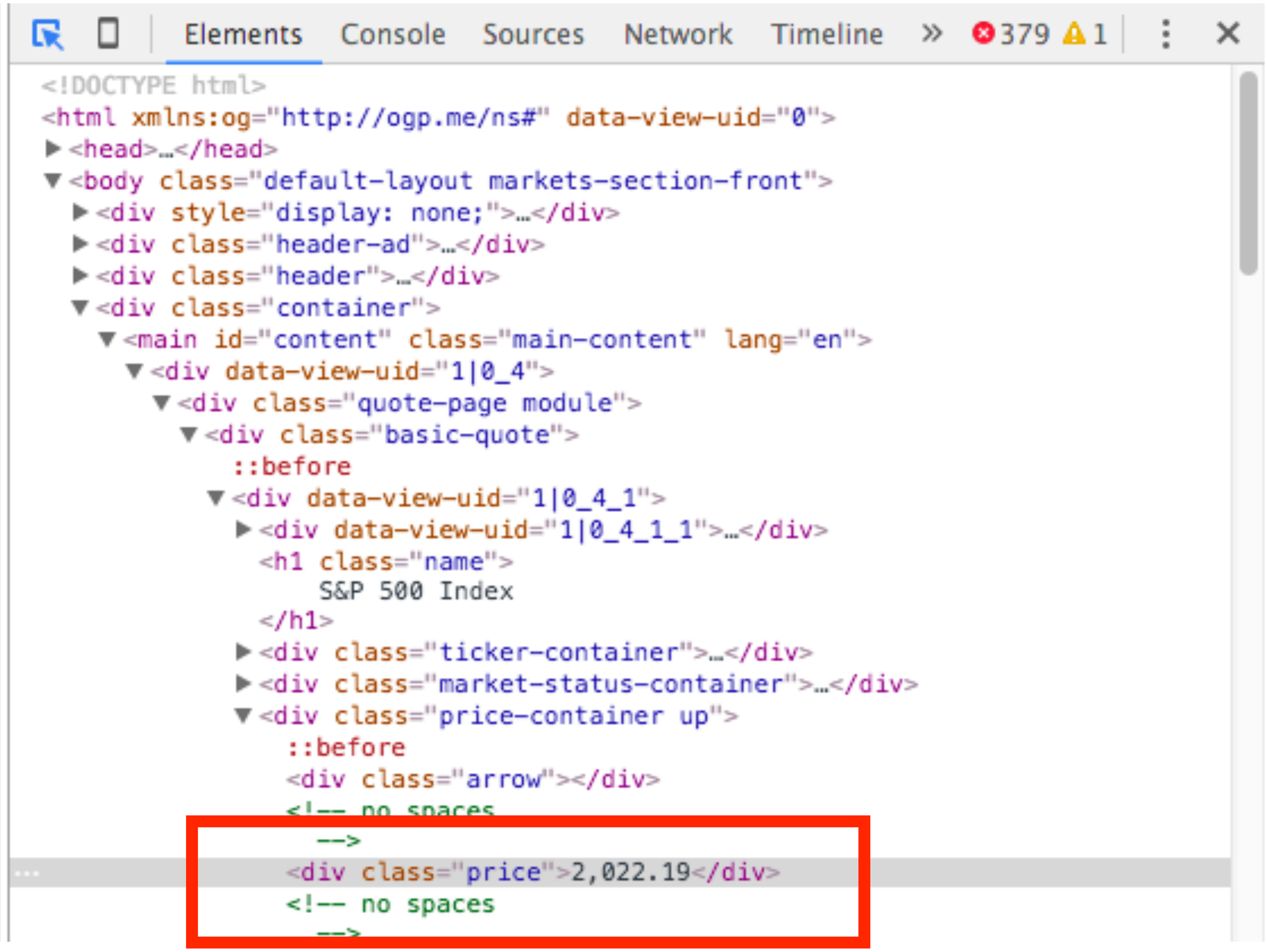
Extracting the stock name



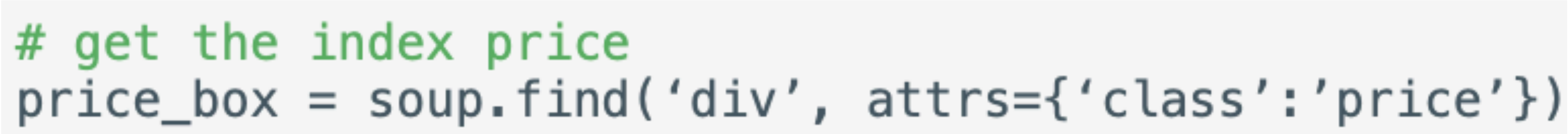
```
<!DOCTYPE html>
<html xmlns:og="http://ogp.me/ns#" data-view-uid="0">
▶<head>…</head>
▼<body class="default-layout markets-section-front">
  ▶<div style="display: none;">…</div>
  ▶<div class="header-ad">…</div>
  ▶<div class="header">…</div>
  ▼<div class="container">
    ▼<main id="content" class="main-content" lang="en">
      ▼<div data-view-uid="1|0_4">
        ▼<div class="quote-page module">
          ▼<div class="basic-quote">
              ::before
            ▼<div data-view-uid="1|0_4_1">
              ▶<div data-view-uid="1|0_4_1_1">…</div>
                <h1 class="name">
                    S&P 500 Index
                </h1>
              ▶<div class="ticker-container">…</div>
              ▶<div class="market-status-container">…</div>
              ▼<div class="price-container up">
                  ::before
                  <div class="arrow"></div>
                  <!-- no spaces
                    -->
                  <div class="price">2,022.19</div>
                  <!-- no spaces
                    -->
```

```
# Take out the <div> of name and get its value
name_box = soup.find(‘h1’, attrs={‘class’: ‘name’})
```

Extracting the stock price

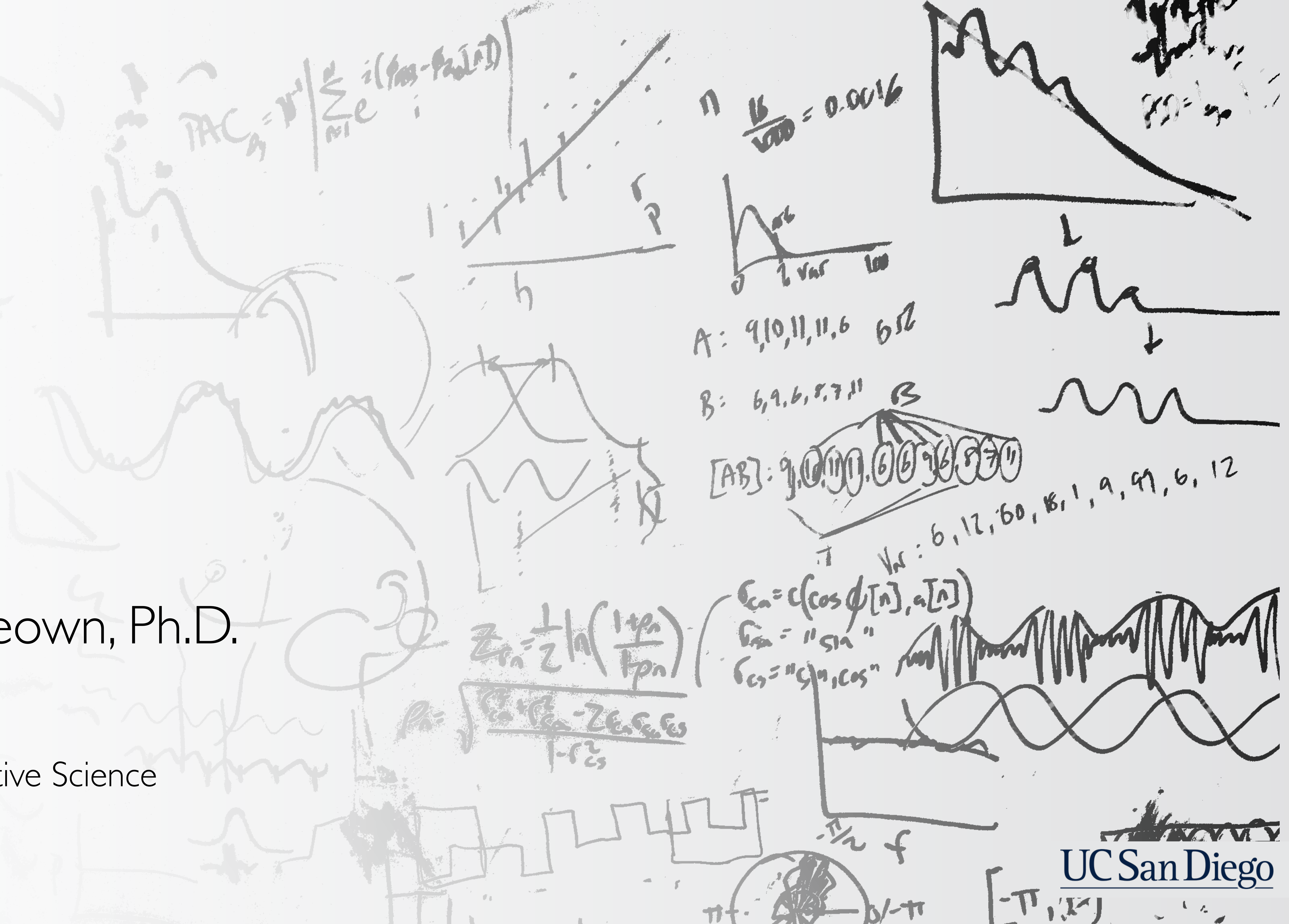


```
<!DOCTYPE html>
<html xmlns:og="http://ogp.me/ns#" data-view-uid="0">
 ▶<head>…</head>
 ▼<body class="default-layout markets-section-front">
   ▶<div style="display: none;">…</div>
   ▶<div class="header-ad">…</div>
   ▶<div class="header">…</div>
   ▼<div class="container">
     ▼<main id="content" class="main-content" lang="en">
       ▼<div data-view-uid="1|0_4">
         ▼<div class="quote-page module">
           ▼<div class="basic-quote">
               ::before
             ▼<div data-view-uid="1|0_4_1">
               ▶<div data-view-uid="1|0_4_1_1">…</div>
                <h1 class="name">
                    S&P 500 Index
                </h1>
               ▶<div class="ticker-container">…</div>
               ▶<div class="market-status-container">…</div>
               ▼<div class="price-container up">
                  ::before
                  <div class="arrow"></div>
                  <!-- no spaces
                    -->
                  <div class="price">2,022.19</div>
                  <!-- no spaces
                    -->
```



```
# get the index price
price_box = soup.find(‘div’, attrs={‘class’:’price’})
```

UC San Diego



Christopher Keown, Ph.D.
UC San Diego

Department of Cognitive Science

ckeown@ucsd.edu

UC San Diego