

Swiss Python Summit Sept. 21 2023

Efficient Transformations with pandas DataFrames: Data for Web Apps

<https://michaelvincerra.github.io> | michael.vincerra@gmail.com

Why use DataFrames to support Web App databases?

- Dataframes:
 - Filter large datasets to only required keys/data structures
 - Decrease data processing time, storage needs
 - Support continuously evolving JSON objects
 - Use Extract Transform Load (ETL) method
 - Database is a static JSON object

oneAPI Samples Browser

for Intel® oneAPI Toolkits

Get started on oneAPI development with code samples

Total Records: 179

▸ Filter 

1D Heat Transfer

The 1D Heat Transfer sample simulates 1D Heat Transfer problem using SYCL

Concepts and
Functionality

cpp

CPU

GPU

AC Fixed

An Intel® FPGA tutorial demonstrating how to use the Algorithmic C Fixed Point Data Type (AC Fixed)

Concepts and
Functionality

cpp

FPGA

AC Int

An Intel® FPGA tutorial demonstrating how to use the Algorithmic C Integer (AC Int)

Concepts and
Functionality

cpp

FPGA

<https://oneapi-src.github.io/oneAPI-samples/>

Business Challenge

Problem statement ¶

- Create a **JSON Object**, where it's:
 - Accessible from Python, HTML, and JavaScript
 - Updated, which triggers a CI script (GitHub Action)
 - Served (for free) on `github.io` pages

Solution

- Use `pandas` dataframes to filter JSON object to smallest fewest lines
- **78% reduction** in lines of code (LOC): Decreased from 12,586 to 2,758 lines JSON file

Create pre-production database

```
def merge_json_files(filepaths:list):
    '''From a list of filenames, output as json a pre-prod database of all sample.json files'''
    results_list = []
    try:
        for f in filepaths:
            with open(f, 'r') as infile:
                results_list.append(json.load(infile))
                # print(results_list)
        with open('sample_db_pre.json', 'w') as output_file:
            json.dump(results_list, output_file)
        return
    except Exception as e:
        print(f"An error occurred. Ensure make_json_list() executes successfully. \n: {e}")
```

Sort and filter database with DataFrames!

```
def df_sort_filter():
    '''Import JSON to DF; sort by name col; filter only records w/ expertise; drop unused columns; add url'''
    start = time.time()

    raw_data = pd.read_json("sample_db_pre.json")
    df = pd.DataFrame(raw_data)
    df = df.sort_values(by=['name'], ignore_index=True, key=lambda x: x.str.lower())
    df = df.dropna(subset=['expertise']) # DROP row if 'expertise' shows "NaN"
    df = df.drop(["guid", "toolchain", "os", "builder", "ciTests", "commonFolder", "dependencies", "category"])
    df['url'] = df.insert(2, 'url', 'np.NaN')
    # print(df)
    end = time.time()
    print(f"Time elapsed:{end-start}")
    return df
```

Reduce lines of code by 78%

Compare how many lines of code in each `JSON` file; before and after applying dataframes.

Pre-production database

```
!bash
```

```
cat sample_db_pre.json | wc -l
```

```
!bash 12585
```

Production database

```
!bash
```

```
cat sample_db_prd.json | wc -l
```

```
!bash 2757
```

Thank you.

Grazie a tutti.

Danke schön.

Merci beaucoup.

Create list of JSON files

```
import os
import json
from collections import OrderedDict
from pathlib import Path
import pandas as pd
import pathlib
import sys
import pprint
pp = pprint.PrettyPrinter(indent=4)
import time
from datetime import datetime

def make_json_list(basedir:str):
    ''' Walk basedir and create list of all filepaths to JSON code samples. '''
    json_file_paths = []
    try:
        if os.path.isdir(basedir):
            for root, dirs, files in os.walk(basedir):
                for file in files:
                    if file.endswith("sample.json"):
                        filepath = os.path.join(root,file)
                        json_file_paths.append(filepath)
            filtered_list = [j for j in json_file_paths if not j.startswith("./oneAPI-samples/Publica
            # print(filtered_list)
            return filtered_list
    except Exception as e:
        print(f"Error. Ensure root directory is oneAPI-samples repo. \n: {e}")
```

Business requirements

Assumptions

- Client requires **no web framework** (not React, Angular, etc.)
- Ability to reproduce app/JSON records **on demand** (daily/quarterly)
- Separate `JSON` files; or a collection of `JSON` files

Requirements

- Create data storage from JSON files where none existed
- Only use JSON objects with "expertise" key
- Implement *stateful* search with "vanilla" JavaScript
- Example `sample.json` file: [INC Quantization for PyTorch](#)

Steps to reproduce

1. Visit [oneAPI Samples](#)
2. `git clone https://github.com/oneapi-src/oneAPI-samples.git`
3. Run this (referenced) jupyter notebook in same directory as cloned repo.

[Efficient DataFrames Notebook on GitHub repo](#)

Summary

Focus: From several standalone `JSON` files, create one database (see #1 below).

oneAPI Samples Browser: 3 main parts

Component	Purpose
1. Python script: <code>db.py</code>	From many <code>JSON</code> files, create one <code>JSON</code> object. Filter and reduce size of object with dataframes
2. Python dev environment, JavaScript, HTML, CSS	Serve a searchable, stateful index and provide UI controls
3. GitHub Action	Create CI script, triggered by changes to application or <code>JSON</code> object

Add URLs for each record

```
def make_url_dict(branch:str,file_paths:list):
    '''Build dict where key is 'name' from sample.json and value is its published url on Github'''
    baseurl = "https://github.com/oneapi-src/oneAPI-samples/tree/{}".format(branch)
    list_name = []
    list_urls = []
    if file_paths is not None:
        for f in file_paths:
            path_base = pathlib.PurePath(f)
            path_slice = '/'.join(path_base.parts[1:-1])
            full_url = os.path.join(baseurl,path_slice)
            list_urls.append(str(full_url))
            with open(f) as file:
                contents = json.load(file)
                tlkt_name = contents["name"]
                list_name.append(tlkt_name)
        raw_dict = dict(zip(list_name,list_urls))
        url_dict = OrderedDict(sorted(raw_dict.items()))
        return url_dict
    else:
        print("An error occurred. Ensure file_paths are generated.")
```

Add URLs to records and transform from DF to JSON

```
def df_add_urls(file_paths:list):
    '''Load all names, urls in make_url_dict(); prep df_sort_filter() and apply urls per their names t
    new_dict = make_url_dict("master", file_paths)
    new_df = df_sort_filter()
    new_df['url'] = new_df['name'].apply(lambda x:new_dict.get(x))
    return new_df

def df_to_db(file_paths:list):
    '''Create prod database, combining df_add_urls(); output for frontend display'''
    df = df_add_urls(file_paths)
    # filepath below must match steps in assoc CI Job, 'with: path: app/dev'
    rev_json = Path('./sample_db_prd.json')
    db = df.to_json(rev_json, orient='records')
    return db
```

main() calls supporting functions

```
%%timeit
print("\n")

def pprinter(json_file):
    with open(file=json_file, mode='r') as read_file:
        json_obj = json.load(read_file)
        obj = json.dumps(json_obj, indent=None)
        # obj.strip("\n")
        pp.pprint(obj)

def main():
    '''Orchestrate sequence of steps to output sample_db_prd.json'''
    rootdir = './'
    file_paths = make_json_list(rootdir)
    merge_json_files(file_paths)
    json_db = df_to_db(file_paths)
    pprinter('sample_db_prd.json')
    with open('sample_db_prd.json', mode='r') as f:
        db = json.load(f)
    print("\nTOTAL RECORDS:", len(db))
    now = datetime.now()
    print(now)
    return json_db
```

if **name** == "**main**": main()

Time elapsed & First 3 records

```
Time elapsed:0.006064653396606445
```

```
('['
```

```
{ "name": "1D Heat Transfer", "description": "The 1D Heat Transfer sample '
'simulates 1D Heat Transfer problem using SYCL", "url": '
'https://github.com/oneapi-src/oneAPI-samples/tree/master/DirectProgramming/C++SYCL/StructuredGrids/1DHeatTransfer
' "languages": [{"cpp": {}}], "targetDevice": ["CPU", "GPU"], "expertise": '
'"Concepts and Functionality"},
```

```
{ "name": "AC Fixed", "description": "An '
'Intel\\u00ae FPGA tutorial demonstrating how to use the Algorithmic C Fixed '
'Point Data Type (AC Fixed) ", "url": '
'https://github.com/oneapi-src/oneAPI-samples/tree/master/DirectProgramming/C++SYCL\_FPGA/Tutorials/FixedPointDataTypes
' "languages": [{"cpp": {}}], "targetDevice": ["FPGA"], "expertise": "Concepts '
'and Functionality"},
```

```
{ "name": "AC Int", "description": "An Intel\\u00ae FPGA '
'tutorial demonstrating how to use the Algorithmic C Integer (AC Int) ", '
'"url": '
'https://github.com/oneapi-src/oneAPI-samples/tree/master/DirectProgramming/C++SYCL\_FPGA/Tutorials/FixedPointDataTypes
' "languages": [{"cpp": {}}], "targetDevice": ["FPGA"], "expertise": "Concepts '
'and Functionality"}
]
```

```
1)
```


fstring.help

Or: How I bought yet another domain

Florian Bruhin



Swiss Python Summit 2023
September 21st

PyFormat

Using `%` and `.format()` for great good!

Contribute on GitHub

Python has had awesome string formatters for many years but the documentation on them is far too theoretic and technical. With this site we try to show you the most common use-cases covered by the old and new style string formatting API with practical examples.

All examples on this page work out of the box with with Python 2.7, 3.2, 3.3, 3.4, and 3.5 without requiring any additional libraries.

Further details about these two formatting methods can be found in the official Python documentation:

- [old style](#)
- [new style](#)

If you want to contribute more examples, feel free to create a pull-request on [Github](#)!

Basic formatting

Simple positional formatting is probably the most common use-case. Use it if the order of your arguments is not likely to change and you only have very few elements you want to concatenate.

Since the elements are not represented by something as descriptive as a name this simple style should only be used to format a relatively small number of elements.

Old

```
'%s %s' % ('one', 'two')
```

New

```
'{} {}'.format('one', 'two')
```

Output

```
o n e   t w o
```

PEP 0498: f-Strings #24

 Open

hut8 opened this issue on Oct 4, 2015 · 11 comments

PEP 498 -- Literal String Interpolation #50

 Open

danielniccoli opened this issue on Dec 13, 2017 · 7 comments

Incorporating f-Strings (PEP 498) #56

 Open

udincer opened this issue on Apr 25, 2019 · 1 comment

Contributions to master, excluding merge commits and bot accounts



60

40

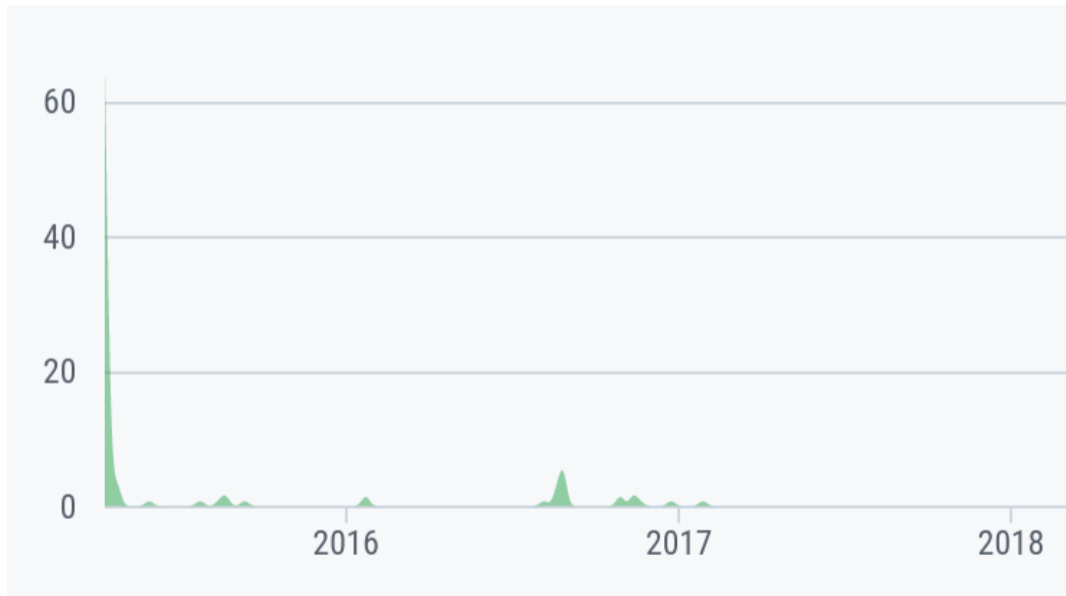
20

0

2016

2017

2018





master ▾



Commits on Feb 4, 2017

Mention v2 in README




ulope committed on Feb 4, 2017 ✖



Commits on Dec 27, 2016

Slightly reword uneven center align padding note

 v2 ▾

 Commits on Jan 16, 2018

Remove Python 3.3 from testing

 **ulope** committed on Jan 16, 2018 

Fix circleci config

 **ulope** committed on Jan 16, 2018 

≈ 2019: fstring.dev?

At PyConDE 2022:
OMG let's do it!
Lightning talks!!!1!

fstring.dev

DEV

Domain not available

Whois

Domain Name: fstring.dev

...

Updated Date: 2022-11-01T17:46:43Z

Creation Date: **2019-09-17**T17:46:43Z

...

Registrant Name: REDACTED FOR PRIVACY

Registrant Organization: **Contact Privacy
Inc. Customer 7151571251**



Unable to load page

Error while opening <http://fstring.dev/>
ERR_NAME_NOT_RESOLVED

[Try again](#)

fstri.ng

Nigeria

99.00 €



Verlängerung: 99.00 €

fstring.cat

CAT

PROMO 5.00 €



~~13.00 €~~

Verlängerung: 25.00 €

Restrictions [\[edit\]](#)

The .cat domain is not territorial, but applies to the whole Catalan-speaking community, whether or not a site is based in Catalonia. In order to be granted a .cat domain, one needs to belong to the Catalan linguistic and cultural community on the Internet. A person, organization or company is considered to belong if they either:^[5]

Pick a language! ▾

NON-STOP NYAN CAT

CREDITS STATS

CHECK OUT NYAN CAT ON FACEBOOK!



GET YOUR YOUTOOZ COLLECTIBLE TODAY!



ADOPT AN OFFICIAL NYAN CAT NET

fstring.help

HELP



32.80 €



Verlängerung: 32.80 €

Python f-strings

All currently supported Python versions (3.6+) support string-formatting via f-strings. While [PEP 498 \(Literal String Interpolation\)](#) as well as the Python documentation ([tutorial](#), [syntax reference](#)) have some information on their usage, I was missing a reference which is terse, but still verbose enough to explain the syntax.

Thus, `fstring.help` was born, made with  and  Jupyter (for now, as a quick hack at PyConDE 2022).

Created by  **BRUHIN**
SOFTWARE

[Consulting, coaching and development for pytest, Qt and Python.](#)

Some content is copied verbatim from [pyformat.info](#) (Copyright 2015 Ulrich Petri, Horst Gutmann).
Thanks!

[Repository](#) on  Github, contributions welcome! If you prefer an interactive version,

 [launch binder](#)

Basic formatting

f-strings are strings with an `f` in front of them: `f"..."`. Inside the f-string, curly braces can be used to format values into it:

```
In [1]: one = 1  
two = 2
```

```
In [2]: f"{one} {two}"
```

```
Out[2]: '1 2'
```

Arbitrary code

You can put any Python code into f-strings:

Python f-string cheat sheets

See [fstring.help](#) for more examples and for a more detailed discussion of this syntax see [this string formatting article](#).

All numbers

The below examples assume the following variables:

```
>>> number = 4125.6
>>> percent = 0.3738
```

Example Output	Replacement Field	Fill	Width	Grouping	Precision	Type
'4125.60'	{number:.2f}				.2	f
'4,125.60'	{number:,.2f}			,	.2	f
'04125.60'	{number:08.2f}	0	8		.2	f
' 4125.60'	{number: 8.2f}		8		.2	f
'37%'	{percent:.0%}				.0	%

These format specifications only work on all numbers (both `int` and `float`).

Recent news

Python 3.7 is EOL: All Python versions support self-documenting expressions:

```
>>> user = 'eric_idle'
```

```
>>> member_since = date(1975, 7, 31)
```

```
>>> f'{user=} {member_since=}'
```

```
"user='eric_idle' member_since=datetime.date(1975, 7, 31)"
```


Recent news

Python 3.7 is EOL: All Python versions support self-documenting expressions:

```
>>> user = 'eric_idle'
>>> member_since = date(1975, 7, 31)
>>> f'{user=} {member_since=}'
"user='eric_idle' member_since=datetime.date(1975, 7, 31)"
```

Python 3.12: PEP 701 – Syntactic formalization of f-strings:

Nested quotes and other valid Python expressions now work inside f-strings:

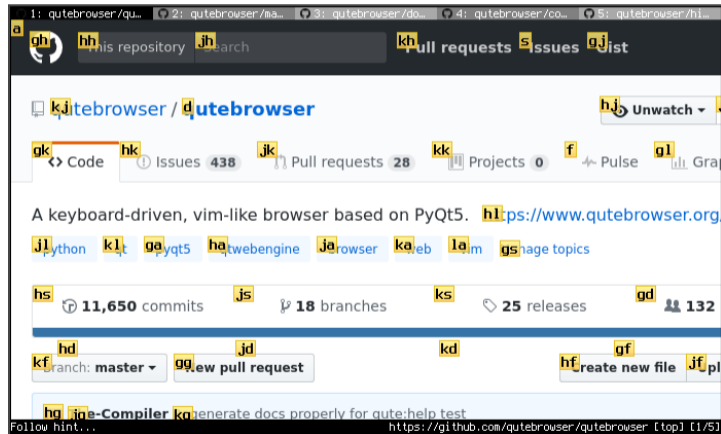
```
>>> f"This is the playlist: {", ".join([
...     'Take me back to Eden', # My, my, those eyes like fire
...     'Alkaline',           # Not acid nor alkaline
...     'Ascensionism'       # Take to the broken skies at last
... ])}"
'This is the playlist: Take me back to Eden, Alkaline, Ascensionism'
```

Shameless plug: qutebrowser!



- Been working on it since 2013
- Python + Qt + QtWebEngine (Chromium)
- I have stickers (for pytest too)!

vim-like browser – like Vimium and friends, but without WebExtension limitations:





<https://fstring.help>

[https://github.com/
The-Compiler/fstring.help](https://github.com/The-Compiler/fstring.help)

https://twitter.com/the_compiler
florian@bruhein.software

Interactive map with statistics

How to visualise Swiss socio-demographic data

Mischa Lisovyi, Python Summit 2023, 21.09.2023



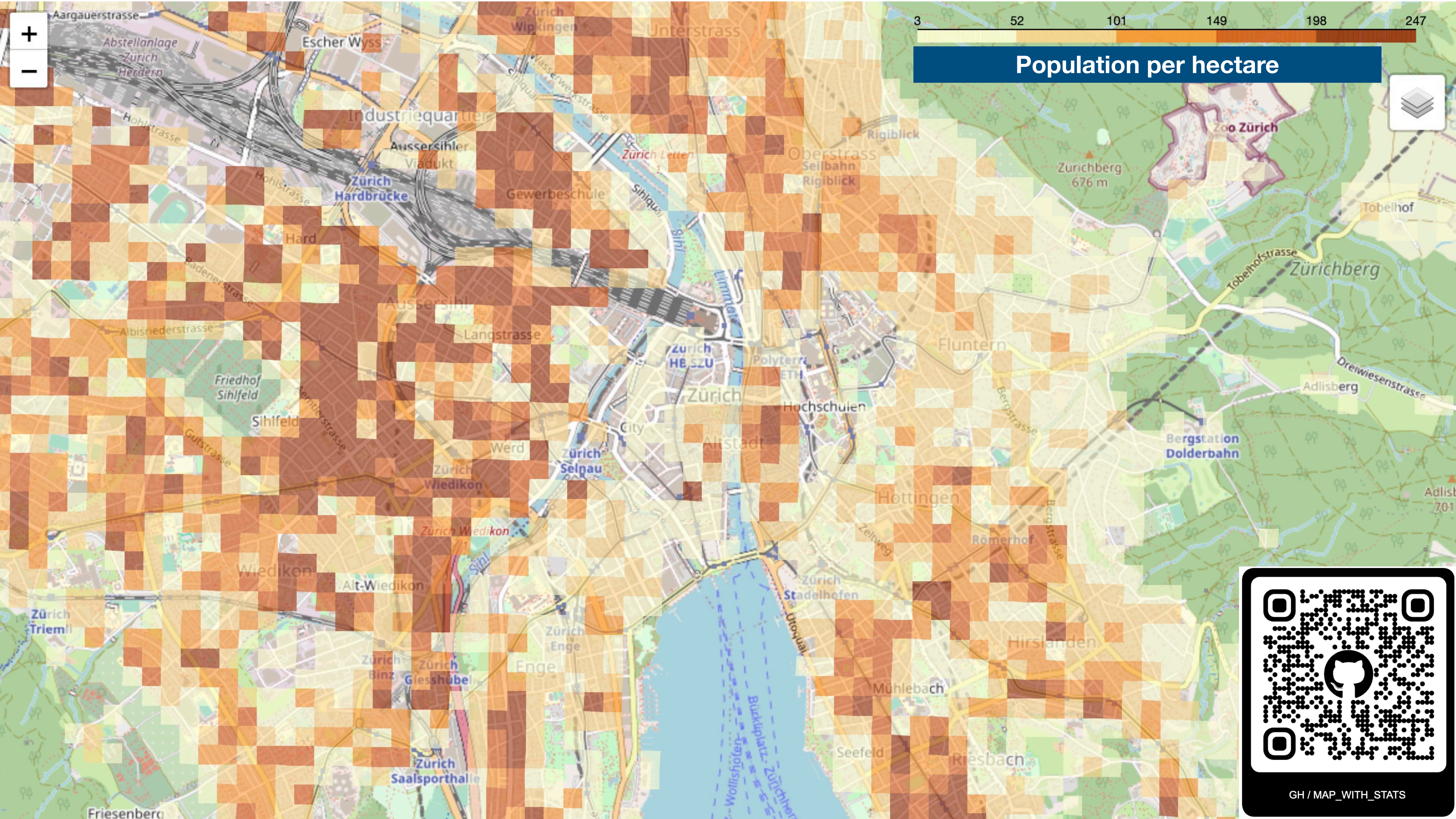


Interactive map with statistics

How to visualise Swiss socio-demographic data

Mischa Lisovyi, Python Summit 2023, 21.09.2023





Population per hectare

3 52 101 149 198 247



How to do it?

1. Get the data

For example:

Bundestamt für Statistik

Population per hectare (100 x 100 meter square)

<https://www.bfs.admin.ch/asset/de/23528269>

The screenshot shows the website of the Swiss Federal Statistical Office (BFS). The header includes the Swiss flag and the text "Schweizerische Eidgenossenschaft", "Confédération suisse", "Confederazione Svizzera", and "Confederaziun svizra". The main navigation bar contains several menu items: "Aktuell", "Statistiken finden", "Dienstleistungen", "Grundlagen und Erhebungen", "Register", "NaDB Nationale Datenbewirtschaftung", "Kompetenzzentrum für Datenwissenschaft", and "Das BFS". A search bar is located in the top right corner with the text "Suchen...".

The breadcrumb trail reads: "Bundestamt für Statistik > Dienstleistungen > GEOSTAT > Geodaten der Bundesstatistik > Gebäude, Wohnungen, Haushalte, Personen > Bevölkerung und Haushalte ab 2010".

The main content area is titled "Statistik der Bevölkerung und Haushalte (STATPOP), Geodaten 2021". On the left side, there is a sidebar with the following items: "Geodaten der Bundesstatistik", "Gebäude, Wohnungen, Haushalte, Personen", "Bevölkerung und Haushalte ab 2010" (highlighted with a red vertical bar), "Gebäude und Wohnungen ab 2010", and "Volks-, Gebäude- und Wohnungszählungen 1970, 1980, 1990, 2000".

Below the title, there is a blue icon representing a map with a location pin. Below the icon, there is a button labeled "Download Publikation (ZIP, 13 MB)". A red arrow points from the QR code on the right towards this button.



How to do it?

1. Get the data
- 2. Install the tool**



```
pip install map_with_stats
```



How to do it?

1. Get the data
2. Install the tool
3. Write python code

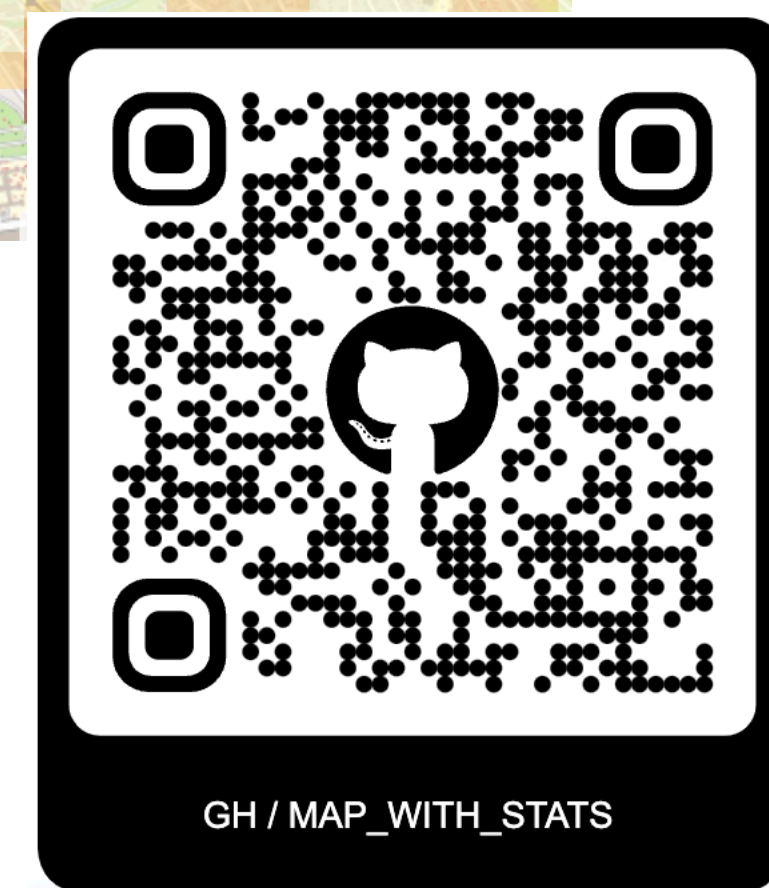
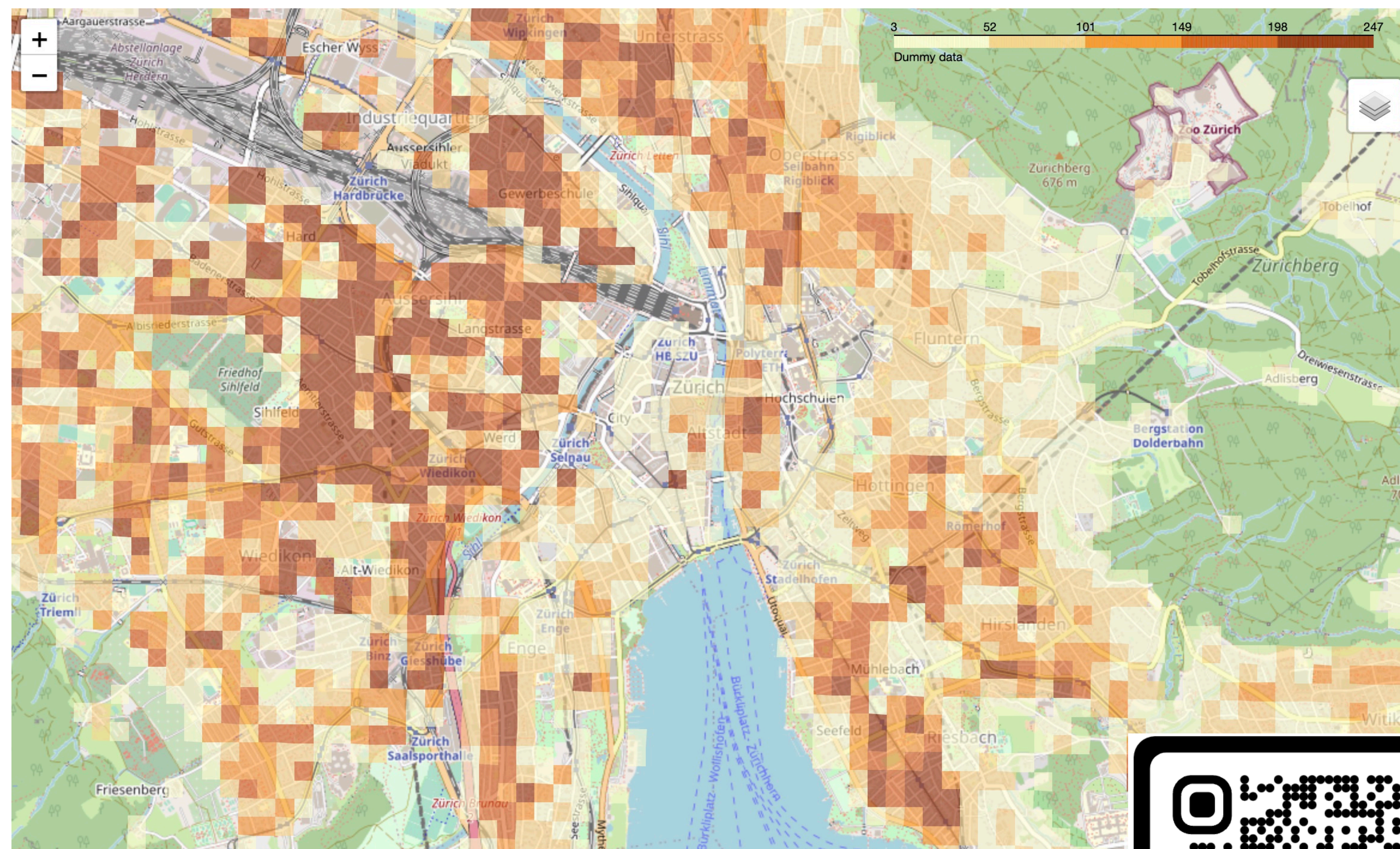


```
1 import pandas as pd
2
3 import map_with_stats as mws
4
5
6 data_raw = pd.read_csv("STATPOP2021.csv", sep=";")
7 data = data_raw[["RELI", "B21BT0T"]]
8 data = mws.hectare2xy(data, "RELI")
9 gdf_stats = mws.create_geo_df_with_hectar_polygons(data, "B21BT0T", crs_out="EPSG:4326")
10
11 # restrict data to the Zürich neighbourhood
12 mask_x = data["X"].between(6700_00, 7000_00)
13 mask_y = data["Y"].between(2330_00, 2630_00)
14 gdf = gdf_stats[mask_x & mask_y]
15
16 # `build_map` is the main helper function that will create a map with a coropleth layer
17 title = "Dummy data"
18 map = mws.build_map(gdf, title, "equidistant")
19 map.save("map.html")
```



How to do it?

1. Get the data
2. Install the tool
3. Write python code
4. **Open map in a browser**



What can you learn from the repository?

If you are junior/middle DS or are curious

- How to **package python code and publish to pypi**
- How to **validate health state** (*pytest + ruff + mypy + pre-commit*)
- How to **enforce code style** (*black + isort + pre-commit*)
- How to **run checks in CI/CD** (*GitHub actions*)
- How to do **documentation** (*mkdocs-material*)



Questions?

Find me @ Apero



Mischa

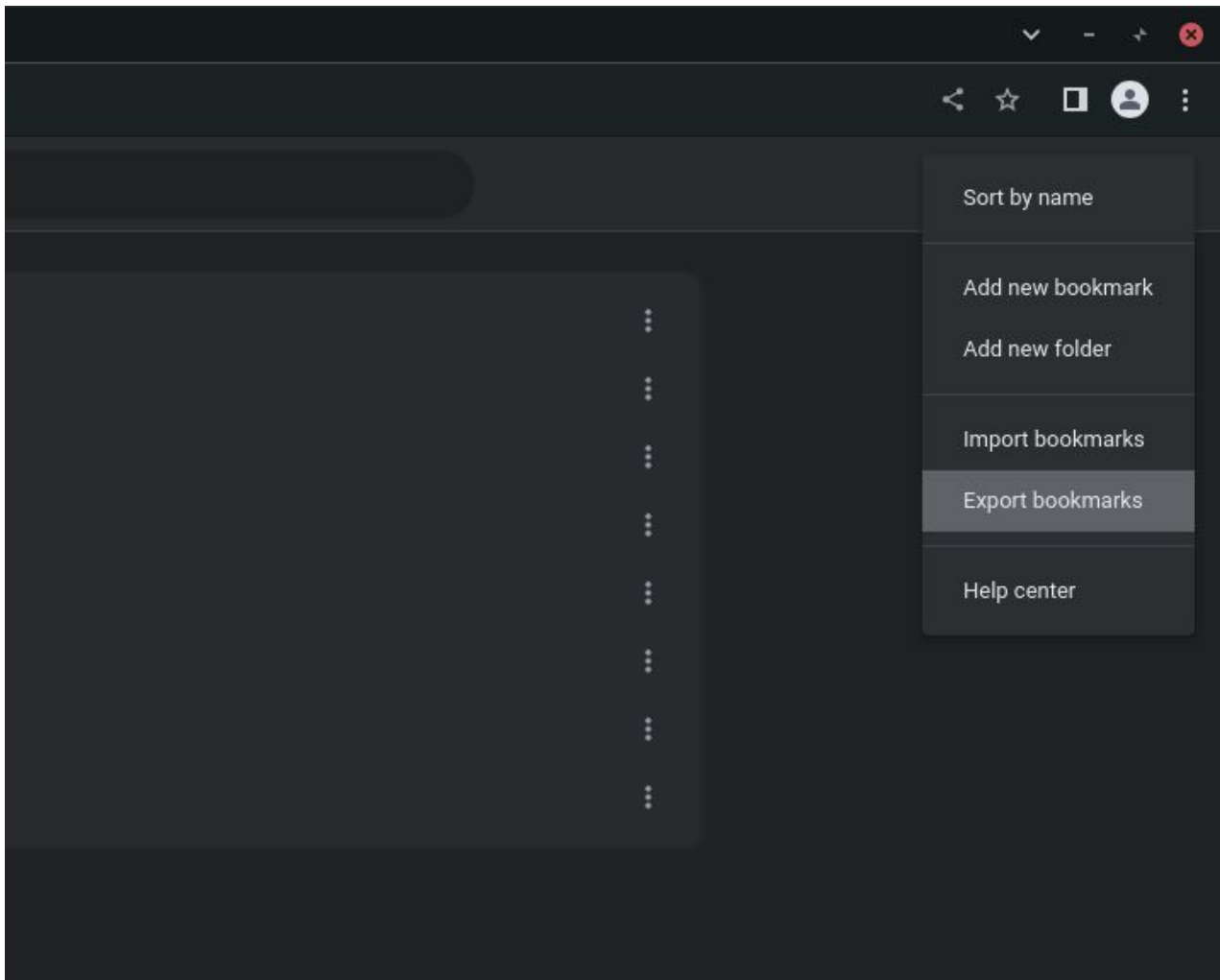


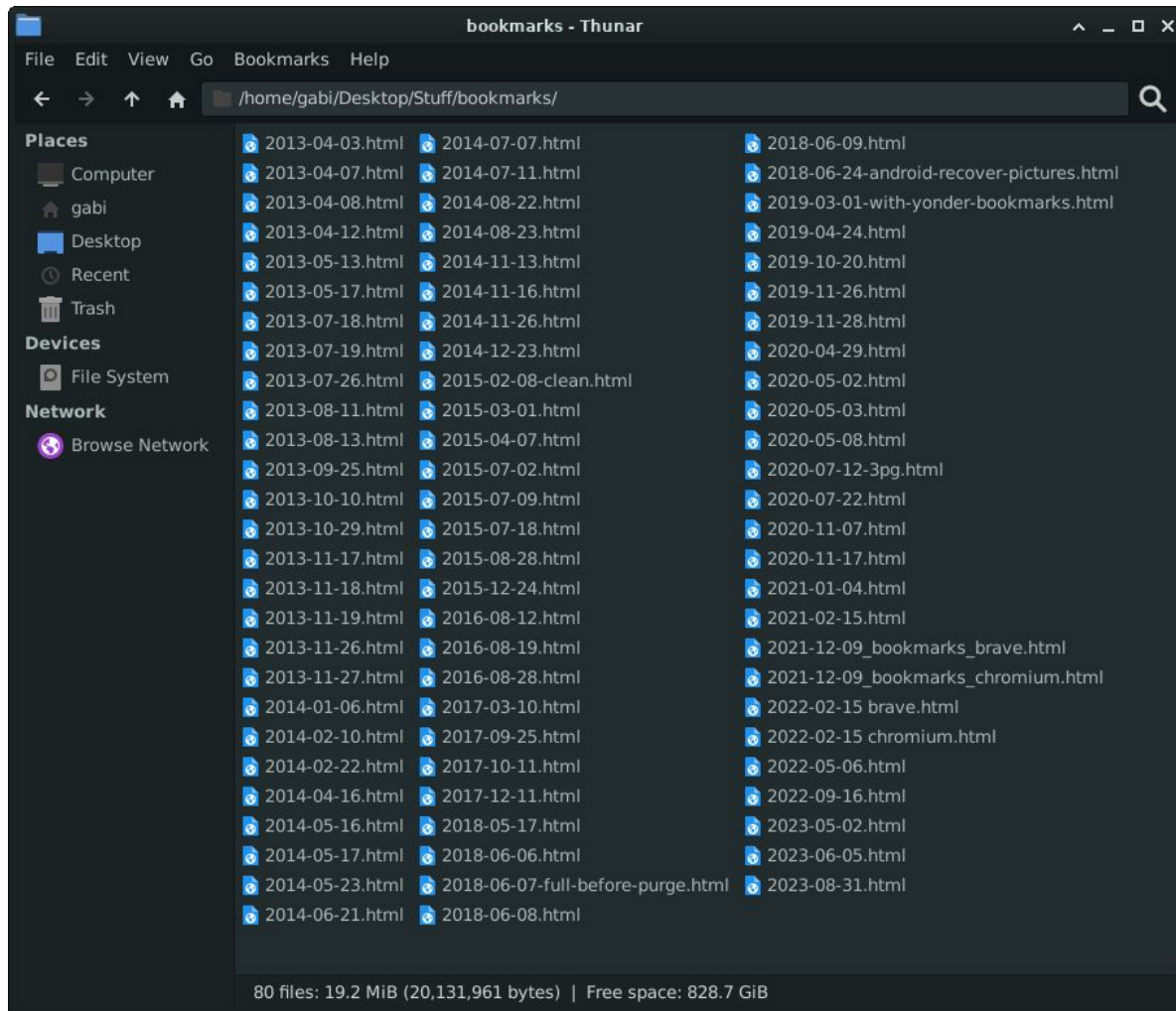


Bookmark DB

Gabi Nagy

github.com/g4b1nagy/bookmark_db





Bookmarks

Bookmarks bar

Old folder A

Old sub-folder C

[Old bookmark C](#)

[Old bookmark A](#)

Old folder B

[Old bookmark B](#)

```
<h3 add_date="0" last_modified="0">0ld folder A</h3>
<dl>
  <p>
  </p>
  <dt>
    <h3 add_date="1693904551" last_modified="1693904780">0ld sub-folder C</h3>
    <dl>
      <p>
      </p>
      <dt><a href="http://old-bookmark-c.db/" add_date="1693904780" icon="data:image/png;base64,iVBORw0KGgoAAAANSUHEUGAAABAAAAAQCAAAAAf8/9hAAADR0LEQVQ4jwVTBwxTZRR+3nvfe2/b3a73rulutxZoN1oYDtyHy3R+sFAGKcuELJQJS0Z/oAYTYqLCL5lXjQka8QMTRAJCBt6jYkJBZnABkJcNGpiSBS06QKT4T4EoZ2Zd03xzxigPcnJ0cmTc56Tn0cwzIIYwMi0SA75UCMKq0IEdyGPHDFcQxY/drSyK7gP/F7J6MA5Ct++n15z7HDvmvTLHygyf487mCAVjPLQzZqG6PefnKHedBRnTbDC/YNwtI8aNz//fsrwLy0Ac0KSGpa1XH5yGmqd08/zRF1nW+3MclhQCQLnt+00q8r2ta+VqPZ36by+3bxQRxHAA1rdw0e0grCgJgAIC9KdKf3n64S7GrBIB0d1lP0nxXAxG7K5CwSKxvirrcKIh+TZduuvSnSZ9W7v6bognDtrwDI5Tb6LYvU6SoEi026Xu8CICYSlizL6pIZbqUk0eqIiGHJM2/tE0Upx7n8T6z9hYfvXQRb51UspX2pTHymSZjJdgCjre0vXew9Rh7h0sULq/P5HC8xAo0xtt2/AChKJl9VADQY/oq8yKfvbsSbm000oM6hlhSHInX5iSfK4DEWTAEOpNAQe600FHkQwAUAAQBTRU6duGIfu50lg6rmpQ/gdwCjgsgLisZ29QUWDFCJsasCA0Z06dcdSjizbNmKrCTbpFNffsiXPhT9Nxpz9gdmm/y1sRVE1D8901x6a/RK2hZtzVrLAXncVHkLJ/Okeosdj3W0TMSKMkmt99Y0cAlJbUqvu276sgTyWsjUakNcKso/vzAzvcej21euz65s2t4CD2Cy+3LAsDU50Sj3lJowZawG5d+HkBJc5yXzY/0b1jNhgr6iP0dF6vDnS3sD83t16tqlk/fmcLJ7Z1sEpVVjZ8CI69ZRebA8RvPfmYRFwDf81m67chJWnj0PDVuf07NblLrb3Eu7f6ojxYfTN1+ctZIsfXbahWbFRgALa5ZPt61p39/7ze0pXuA2j/4YnhHPPnKoEVPVCACVBxtmGpjhEWiEax6igliAQDZi4opGK7NR6ofyRnllb0/4CkNmqZFmGDBNen4/wzBNEw+5dEzCVxAsAIgDsApgH8Kcu0E95AZsXhkTHHcB/2VMnx4gRnQ0AAAAASUUVORK5CYII=">0ld bookmark C</a>
      </dt>
    </dl>
  </p>
  </p>
</dt>
<dt><a href="http://old-bookmark-a.db/" add_date="0">0ld bookmark A</a>
</dt>
</dl>
<p>
```

```
<h3 add_date="0" last_modified="0">old folder A</h3>
<dt>
  <p>
  </p>
</dt>
<h3 add_date="1693904551" last_modified="1693904780">old sub-folder C</h3>
<dt>
  <p>
  </p>
  <dt><a href="http://old-bookmark-c.db/" add_date="1693904780" icon="data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAABAAAAAQCAAAAAf8/9hAAADR0lEQVQ4jWVTbWxTZRR+3nvfe2/b3a73ruIutXZoN1oYDtyHy3R+sFAGKcuELJQJS0Z/oAYTYqLCL5LxjQka8QMTRAJCBt6jYkJBZnABkJcNGpiSBS06QKT4T4EoZ2Zd03xzxIGPcnJ0cmTc56Tn0cwzIIYwMi0SA75UCMKq0IEdyGPHDFcQxY/drSyK7gP/F7J6NA5Ct++n15z7HDvmvTLHyqyf487mCAVjPLQzZqG6PefnKHedBRnTbDC/YNwtI8aNz//fsrwLy0Ac0KSFgpa1XH5yGldqmd08/zRF1nW+3MclhQCQLNt+00q8r2ta+VqPZ36by+3bxQRxHAA1rdw0e0grCgJgAIC9KdKf3n64S7GrBIB0d1lP0nxXAxG7K5CwSKxviircKIh+TZduuvSnS29W7v6bognDtrwDI5Tb6LYvU6SoEi026Xu8CICYSlizL6pIZbqU0eqIiGHjM2/tE0UpX7n8T6z9hYfvXQRb51UspX2pTHymS2jJdgCjre0vXew9Rh7h0sULq/P5HC8xAo0xtt2/AChKJl9VADQY/oq8yKfvsSbm000oM6hlhSHInX5iSfK4DEWTAEOpNAQe600FHkQwAUAAQBTRU6duGI5U0lg6rmpQ/gdwCjgsgLisZ29QUWDFCJsasCA0Z06dcdSjizbNmKrCTbpFNffsIXPhT9NxiPz9gdmm/y1sRVEiD8901x6a/RK2hZtzVrLAXncVHkLJ/0keosdj3W0TMSKMmt99Y0cAlJbUquv276sgTyWsjUAKNcKso/vzAzvcej21euz65s2t4CD2Cy+3LAsDU50Sj3LJoWzawG5d+HKBjc5yXzY/0b1jMhgr6iP0dF6vDnS3sD83t16tqlk/fmcLJ7Z1sEpVVjZ8CI69ZRebA8RvPfmYRFwDf81m67chJWnj0PDVuf07NblLRb3Eu7f6ojxYfTN1+ctZIsfXbahWbFRgALa5ZPt61p39/7ze0pXuA2j/4YnhHPPnKoEPVCACVBxbtmGpJhEWiEax6igliAQDZ14opGK7NR6ofYRnllb0/4CkNmQZFMgDBNen4/wzBNEw+5dEZcVxAsAIGDsApgH8Kcu0E95AZSxhKTHHcB/2VMnx4gRnQ0AAAAASUVR0RK5CYII=">old bookmark C</a>
  </dt>
</dt>
  <p>
  </p>
</dt>
<dt><a href="http://old-bookmark-a.db/" add_date="0">old bookmark A</a>
</dt>
</dt>
<p>
```

```
<h3 add_date="0" last_modified="0">old folder A</h3>
<dt>
  <p>
  </p>
</dt>
<dt>
  <h3 add_date="1693904551" last_modified="1693904780">old sub-folder C</h3>
  <dt>
    <p>
    </p>
    <dt><a href="http://old-bookmark-c.db/" add_date="1693904780" icon="data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAABAAAAAQCAAAAAf8/9hAAADR0lEQVQ4jWVTbWxTZRR+3nvfe2/b3a73rulutXZoN1oYDtyHy3R+sFAGKcuELJQJS0Z/oAYTYqLCL5LxjQka8QMTRAJCZBt6jYkJBZnABkJcNGpiSBS06QKT4T4EoZ2Zd03xzxigPcnJ0cmTc56Tn0cWzIIYwMi0SA75UCMKq0IEdyGPHDFcQxY/drSyK7gP/F7J6NA5Ct++n15z7HDvmvTLHyqyf487mCAVjPLQzZqG6PefnKHedBRnTbDC/YNwtI8aNZ//fsrwLy0Ac0KSFgpa1XH5yGldmqd08/zRF1nW+3MclhQCQLnt+00q8r2ta+VqPZ36by+3bxQRxHAA1rdw0e0grCgJgAIC9KdKf3n64S7GrBIB0d1lP0nxXAxG7K5CwSKxviircKIh+TZduuvSnS29W7v6bognDtrwDI5Tb6LYvU6SoEi026Xu8CICYSlizL6pIZbqUk0eqIiGHjM2/tE0UpX7n8T6z9hYfvXQRb51UspX2pTHymS2jJdgCjre0vXew9Rh7h0sULq/P5HC8xAo0xtt2/AChKJl9VADQY/oq8yKfvbsSbm000oM6hlhSHInX5iSfK4DEWTAEOpNAQe600FHkQwAUAAQBTRU6duGI5U0lg6rmpQ/gdwCjgsgLisZ29QUWDFCJsasCA0Z06dcdSjizbNmKrCTbpFNffsIXPhT9NxiPz9gdmm/y1sRVEiD8901x6a/RK2hZtzVrLAXncVHkLJ/0keosdj3W0TMSKMKmt99Y0cAlJbUqvU276sgTyWsjUAKNcKso/vzAzvcej21euz65s2t4CD2Cy+3LAsDU50Sj3LJoWzawG5d+HKbjc5yXzY/0b1jMhgr6iP0dF6vDnS3sD83t16tqlk/fmcLJ7Z1sEpVVjZ8CI69ZRebA8RvPfmYRFwDf81m67chJWnj0PDVuf07NblLRb3Eu7f6ojxYfTN1+ctZIsfXbahWbFRgALa5ZPt61p39/7ze0pXuA2j/4YnhHPPnKoEPVCACVBxbtmGpjhEWiEax6igliAQDZ14opGK7NR6ofYRnllb0/4CkNmQZFMgDBNen4/wzBNEw+5dEZcVxAsA1gDsApgH8Kcu0E95AZsXhKTHHcB/2VMnx4gRnQ0AAAAASUVORK5CYII=">old bookmark C</a>
    </dt>
  </dt>
  <p>
  </p>
</dt>
<dt><a href="http://old-bookmark-a.db/" add_date="0">old bookmark A</a>
</dt>
</dt>
<p>
```

```
<h3 add_date="0" last_modified="0">old folder A</h3>
<dt>
  <p>
  </p>
</dt>
<h3 add_date="1693904551" last_modified="1693904780">old sub-folder C</h3>
<dt>
  <p>
  </p>
  <dt><a href="http://old-bookmark-c.db/" add_date="1693904780" icon="data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAAABAAAAAQCAAAAAf8/9hAAADR0lEQVQ4jWVTbWxTZRR+3nvfe2/b3a73rulutXZoNlOyDtyHy3R+sFAGKcuELJQJS0Z/oAYTYqLCL5LxjQka8QMTRAJCBt6jYkJBZnABkJcNGpiSBS06QKT4T4Eoz2Zd03xzxigPcnJ0cmTc56Tn0cwzIIYwMi0SA75UCMKq0IEdyGPHDFcQxY/drSyK7gP/F7J6NA5Ct++n15z7HDvmvTLHyqyf487mCAVjPLQzZqG6PefnKHedBRnTbDC/YNwtI8aNz//fsrwLy0Ac0KSFgpa1XH5yGldmd08/zRF1nW+3MclhQCQLnt+00q8r2ta+VqPZ36by+3bxQRxHAA1rdw0e0grCgJgAIC9KdKf3n64S7GrBIB0d1lP0nxXAxG7K5CwSKxviircKIh+TZduuvSnS29W7v6bognDtrwDI5Tb6LYvU6SoEi026Xu8CICYSlizL6pIZbquk0eqIiGHjM2/tE0UpX7n8T6z9hYfvXQRb51UspX2pTHymSZjJdgCjre0vXew9Rh7h0sULq/P5HC8xAo0xtt2/AChKJl9VADQY/oq8yKfvsSbm000oM6hlhSHInX5iSfK4DEWTAEOpNAQe600FHkQwAUAAQBTRU6duGI5U0lg6rmpQ/gdwCjgsgLis2Z9QUWDFCJsasCA0Z06dcdSjizbNmKrCTbpFNffsIXPhT9NxiPz9gdmm/y1sRVEiD8901x6a/RK2hZtzVrLAXncVHkLJ/0keosdj3W0TMSKMKmt99Y0cAlJbUquv276sgTyWsjUAKNcKso/vzAzvcej21euz65s2t4CD2Cy+3LAsDU50Sj3lJoWzawG5d+HKBjc5yXzY/0b1jMhgr6iP0dF6vDnS3sD83t16tqlk/fmcLJ7Z1sEpVVjZ8CI69ZRebA8RvPfmYRFwDf81m67chJWnj0PDVuf07NblLRb3Eu7f6ojxYfTN1+ctZIsfXbahWbFRgALa5ZPt61p39/7ze0pXuA2j/4YnhHPPnKoEVPVACVBxbtmGpjhEWiEax6igliAQDZ14opGK7NR6ofYRnllb0/4CkNmQZFMgDBNen4/wzBNEw+5dEZcVxAsAIGDsApgH8Kcu0E95AZsXhThHHcB/2VMnx4gRnQ0AAAAASUVR0K5CYII=">old bookmark C</a>
  </dt>
</dt>
  <p>
  </p>
</dt>
<dt><a href="http://old-bookmark-a.db/" add_date="0">old bookmark A</a>
</dt>
</dt>
<p>
```

```
<h3 add_date="0" last_modified="0">old folder A</h3>
<dt>
  <p>
  </p>
</dt>
<h3 add_date="1693904551" last_modified="1693904780">old sub-folder C</h3>
<dt>
  <p>
  </p>
  <dt><a href="http://old-bookmark-c.db/" add_date="1693904780" icon="data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAABAAAAAQCAAAAAf8/9hAAADR0lEQVQ4jWVTbWxTZRR+3nvfe2/b3a73rulutXZoN1oYDtyHy3R+sFAGKcuELJQJS0Z/oAYTYqLCL5LxjQka8QMTRAJCBt6jYkJBznABkjcNGpiSBS06QKT4T4Eoz2Zd03xzxigPcnJ0cmTc56Tn0cwzIIYwMi0SA75UCMKq0IEdyGPHDFcQxY/drSyK7gP/F7J6NA5Ct++n15z7HDvmvTLHyqyf487mCAVjPLQzZqG6PefnKHedBRnTbDC/YNwtI8aNZ//fsrwLy0Ac0KSFgpa1XH5yGldqmd08/zRF1nW+3MclhQCQLnt+00q8r2ta+VqPZ36by+3bxQRxHAA1rdw0e0grCgJgAIC9KdKf3n6457GrBIB0d1lP0nxXAxG7K5CwSKxviircKIh+TZduuvSnS29W7v6bognDtrwDI5Tb6LYvU6SoEi026Xu8CICYSlizL6pIZbqU0eqIiGHjM2/tE0UpX7n8T6z9hYfvXQRb51UspX2pTHymSZjJdgCjre0vXew9Rh7h0sULq/P5HC8xAo0xtt2/AChKJl9VADQY/oq8yKfvsSbm000oM6hlhSHInX5iSfK4DEWTAEOpNAQe600FHkQwAUAAQBTRU6duGIffU50lg6rmpQ/gdwCjgsgLis2Z9QUWDFCJsasCA0Z06dcdSjizbNmKrCTbpFNffsIXPhT9NxiPz9gdmm/y1sRVEiD8901x6a/RK2hZtzVrLAXncVHkLJ/0keosdj3W0TMSKMKmt99Y0cAlJbUqvu276sgTyWsjUAKNcKso/vzAzvcej21euz65s2t4CD2Cy+3LAsDU50Sj3lJowZawG5d+HKbjc5yXzY/0b1jMhgr6iP0dF6vDnS3sD83t16tqlk/fmcLJ7Z1sEpVVjZ8CI69ZRebA8RvPfmYRFwDf81m67chJWnj0PDVuf07NblLRb3Eu7f6ojxYfTN1+ctZIsfXbahWbFRgALa5ZPt61p39/7ze0pXuA2j/4YnhHPPnKoEVPVACVxbtmgPjhEWiEax6igliAQDZ14opGK7NR6ofYRnllb0/4CkNmQZFMgDBNEn4/wzBNEw+5dEZcVxAsAIGDsApgH8Kcu0E95AZsXhkTHHcB/2VMnx4gRnQ0AAAAASUVR0K5CYII=">old bookmark C</a>
  </dt>
</dt>
  <p>
  </p>
</dt>
<dt><a href="http://old-bookmark-a.db/" add_date="0">old bookmark A</a>
</dt>
</dt>
<p>
```

```
<h3 add_date="0" last_modified="0">old folder A</h3>
<dt>
  <p>
  </p>
</dt>
<h3 add_date="1693904551" last_modified="1693904780">old sub-folder C</h3>
<dt>
  <p>
  </p>
  <dt><a href="http://old-bookmark-c.db/" add_date="1693904780" icon="data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAABAAAAAQCAAAAAf8/9hAAADR0lEQVQ4jWVTbWxTZRR+3nvfe2/b3a73rulutxZoN1oYDtyHy3R+sFAGKcuELJQJS0Z/oAYTYqLCL5LxjQka8QMTRAJCBt6jYkJBZnABkJcNGpiSBS06QKT4T4Eoz2Zd03xzxiGPcnJ0cmTc56Tn0cwzIYwMi0SA75UCMKq0IEdyGPHDFcQxY/drSyK7gP/F7J6NA5Ct++n15z7HDvmvTLHyqyf487mCAVjPLQzZqG6PefnKHedBRnTbDC/YNwtI8aNZ//fsrwLy0Ac0KSFgpa1XH5yGldqmd08/zRF1nW+3MclhQCQLnt+00q8r2ta+VqPZ36by+3bxQRxHAA1rdw0e0grCgJgAIC9KdKf3n6457GrBIB0d1lP0nxXAxG7K5CwSKxviircKIh+TZduuvSnS29W7v6bognDtrwDI5Tb6LYvU6SoEi026Xu8CICYSlizL6pIZbqu0eqIiGHjM2/tE0Upx7n8T6z9hYfvXQRb51UspX2pTHymSZjJdgCjre0vXew9Rh7h0sULq/P5HC8xAo0xtt2/AChKJl9VADQY/oq8yKfvbsSbm000oM6hlhSHInX5iSfK4DEWTAEOpNAQe600FHkQwAUAAQBTRU6duGI5U0lg6rmpQ/gdwCjgsgLisZ29QUWDFCJsasCA0Z06dcdSjizbNmKrCTbpFNffsIXPhT9NxiPz9gdmm/y1sRVEiD8901x6a/RK2hZtzVrLAXncVHkLJ/0keosdj3W0TMSKMkmt99Y0cAlJbUquv276sgTyWsjUAKNcKso/vzAzvcej21euz65s2t4CD2Cy+3LAsDU50Sj3LJoWzawG5d+HKBjc5yXzY/0b1jMhgr6iP0dF6vDnS3sD83t16tqlk/fmcLJ7Z1sEpVVjZ8CI69ZRebA8RvPfmYRFwDf81m67chJWnj0PDVuf07NblLRb3Eu7f6ojxYfTN1+ctZIsfXbahWbFRgALa5ZPt61p39/7ze0pXuA2j/4YnhHPPnKoEVPVCACVBxbtmGpjhEWiEax6igliAQDZ14opGK7NR6ofYRnllb0/4CkNmQZFMgDBNen4/wzBNEw+5dEZcVxAsAIGDsApgH8Kcu0E95AZsXhkThHHcB/2VMnx4gRnQ0AAAAASUUVORK5CYII=">old bookmark C</a>
  </dt>
</dt>
  <p>
  </p>
</dt>
<dt><a href="http://old-bookmark-a.db/" add_date="0">old bookmark A</a>
</dt>
</dt>
<p>
```

```
<h3 add_date="0" last_modified="0">old folder A</h3>
<dt>
  <p>
  </p>
</dt>
<h3 add_date="1693904551" last_modified="1693904780">old sub-folder C</h3>
<dt>
  <p>
  </p>
  <dt><a href="http://old-bookmark-c.db/" add_date="1693904780" icon="data:image/png;base64,iVBORw0KGGoAAAANSUHEUGAAABAAAAAQAYAAAAf8/9hAAADR0lEQVQ4jWVTbWxTZRR+3nvfe2/b3a73rulutxZoNl0yDtyHy3R+sFAGKcuELJQJS0Z/oAYTYqLCL5LxjQka8QMTRAJCBt6jYkJBZnABkJcNGpiSBS06QKT4T4EoZ2Zd03xzxiGPcnJ0cmTc56Tn0cwzIYwMi0SA75UCMKq0IEdyGPHDFcQxY/drSyK7gP/F7J6NA5Ct++n15z7HDvmvTLHyqyf487mCAVjPLQzZqG6PefnKHedBRnTbDC/YNwtI8aNZ//fsrwLy0Ac0KSFgpa1XH5yGldqmd08/zRF1nW+3MclhQCQLnt+00q8r2ta+VqPZ36by+3bxQRxHAA1rdw0e0grCgJgAIC9KdKf3n64576rBIB0d1lP0nxXAxG7K5CwSKxviircKIh+TZduuvSnS29W7v6bognDtrwDI5Tb6LYvU6SoEi026Xu8CICYSlizL6pIZbqu0eqIiGHjM2/tE0Upx7n8T6z9hYfvXQRb51UspX2pTHymSZjJdgCjre0vXew9Rh7h0sULq/P5HC8xAo0xtt2/AChKJl9VADQY/oq8yKfvsSbm000oM6hlhSHInX5iSfK4DEWTAEOpNAQe600FHkQwAUAAQBTRU6duGI5U0lg6rmpQ/gdwCjgsgLisZ29QUWDFCJsasCA0Z06dcdSjizbNmKrCTbpFNffsIXPhT9NxiPz9gdmm/y1sRVEiD8901x6a/RK2hZtzVrLAXncVHkLJ/0keosdj3W0TMSKMkmt99Y0cALJbUquv276sgTyWsjUAKNcKso/vzAzvcej21euz65s2t4CD2Cy+3LAsDU50Sj3LJoWzawG5d+HKBjc5yXzY/0b1jMhgr6iP0dF6vDnS3sD83t16tqlk/fmcLJ7Z1sEpVVjZ8CI69ZRebA8RvPfmYRFwDf81m67chJWnj0PDVuf07NblLrb3Eu7f6ojxYfTN1+ctZIsfXbahWbFRgALa5ZPt61p39/7ze0pXuA2j/4YnhHPPnKoEVPVACVxbtmgPjHEWiEax6igliAQDZ14opGK7NR6ofyRnllb0/4CkNmqZFmQDBNen4/wzBNEw+5dEZcVxAsAIGDsApgH8Kcu0E95AZsXhkTHHcB/2VMnx4gRnQ0AAAAASUUVORK5CYII=">old bookmark C</a>
  </dt>
</dt>
  <p>
  </p>
</dt>
<dt><a href="http://old-bookmark-a.db/" add_date="0">old bookmark A</a>
</dt>
</dt>
<p>
```




Libraries



Libraries

- Beautiful Soup



Libraries

- Beautiful Soup (html5lib)



Libraries

- Beautiful Soup (html5lib)
- Django
















Code

```
def parse_html(html, bookmarks=None, labels=None, current_labels=None):  
    """  
    1. Because bookmark folders can be nested,  
       a recursive algorithm is required to parse them out.  
    2. Still confused? See point 1.  
    """
```

Select bookmark to change

ADD BOOKMARK  Search 13 results (4553 total)Action: Go 0 of 13 selected

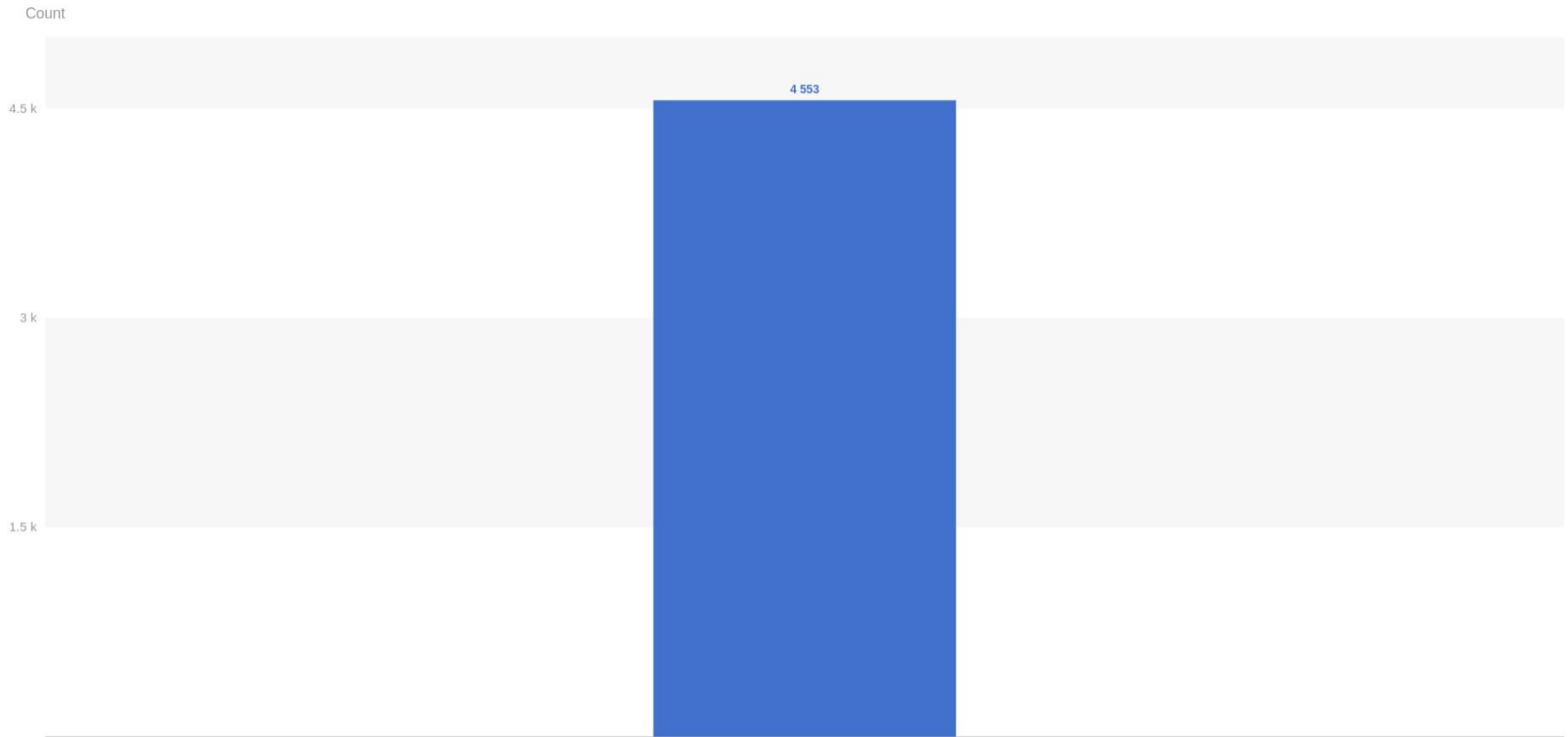
<input type="checkbox"/>	ICON	CREATED ON	NAME	URL
<input type="checkbox"/>		2018-06-23 13:57	DigDeep Image Recovery - Apps on Google Play	https://play.google.com/store/apps/details?id=com.greatstuffapps.digdeep&hl=en
<input type="checkbox"/>		2018-06-23 13:57	DiskDigger photo recovery - Apps on Google Play	https://play.google.com/store/apps/details?id=com.defianttech.diskdigger
<input type="checkbox"/>		2018-06-23 13:57	Undeleter Recover Files & Data - Apps on Google Play	https://play.google.com/store/apps/details?id=fahrbot.apps.undelete
<input type="checkbox"/>		2018-06-23 13:41	[APP][ROOT] USB MASS STORAGE Enabler v1.6 Android Development and Hacking	https://forum.xda-developers.com/android/apps-games/app-universal-mass-storage-enabler-beta-t3240097
<input type="checkbox"/>		2018-06-23 13:41	data recovery - How to recover deleted photos from Samsung phone - Super User	https://superuser.com/questions/860201/how-to-recover-deleted-photos-from-samsung-phone
<input type="checkbox"/>		2018-06-23 13:41	How to change the USB connection options on my Motorola Moto G (3rd Generation)	https://support.bell.ca/Mobility/Smartphones_and_mobile_internet/Motorola-Moto-G.how_to_change_the_usb_connection_options_on_my
<input type="checkbox"/>		2018-06-23 13:41	PhotoRec FAQ - CGSecurity	https://www.cgsecurity.org/wiki/PhotoRec_FAQ#How_to_enable_the_USB_mass_storage_mode
<input type="checkbox"/>		2018-06-23 13:41	mtp - How to mount Smartphone to recover files - Ask Ubuntu	https://askubuntu.com/questions/446180/how-to-mount-smartphone-to-recover-files
<input type="checkbox"/>		2018-06-23 13:41	[Data Recovery] My phone is rooted, but my data recovery program says otherwise : Nexus5	https://www.reddit.com/r/Nexus5/comments/5p7str/data_recovery_my_phone_is_rooted_but_my_data/
<input type="checkbox"/>		2018-06-23 13:41	[Help] How to recover data After factory reset : GalaxyS6	https://www.reddit.com/r/GalaxyS6/comments/4jlixn/help_how_to_recover_data_after_factory_reset/
<input type="checkbox"/>		2018-06-23 13:41	[OFFICIAL] Mac Android Data Recovery: Recover Deleted/Lost Data on Android	https://drfone.wondershare.com/mac-android-data-recovery.html
<input type="checkbox"/>		2018-06-23 13:41	Free Android Data Recovery Software - EaseUS MobiSaver Free for Android.	https://www.easeus.com/android-data-recovery-software/free-android-data-recovery.html
<input type="checkbox"/>		2018-06-23 13:41	Android File Recovery - Get Files Back from Android Phone/Tablets	https://www.fonepaw.com/tutorials/android-file-recovery.html#solution3

13 bookmarks

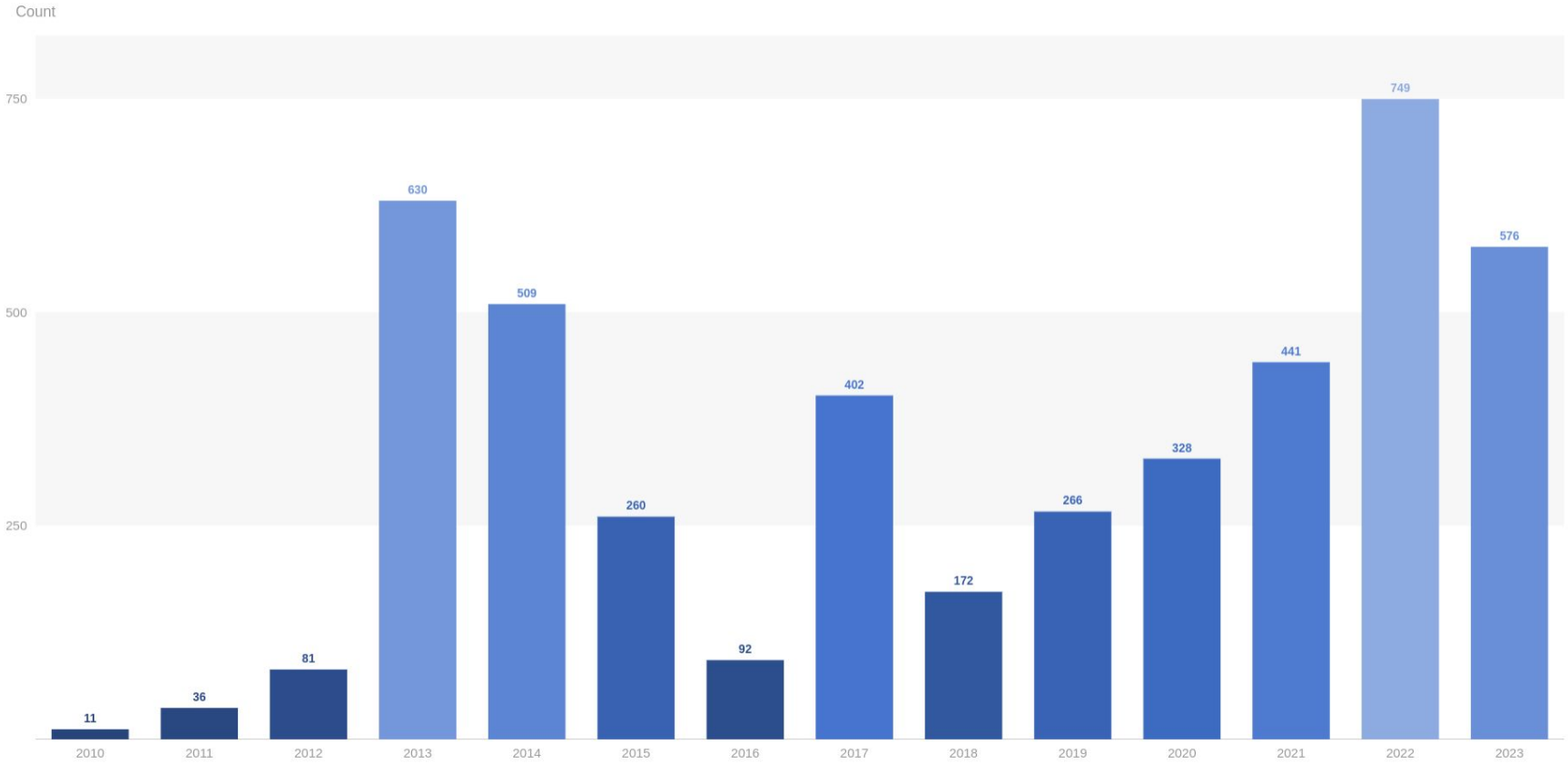
VIZZU

vizzuhq.com

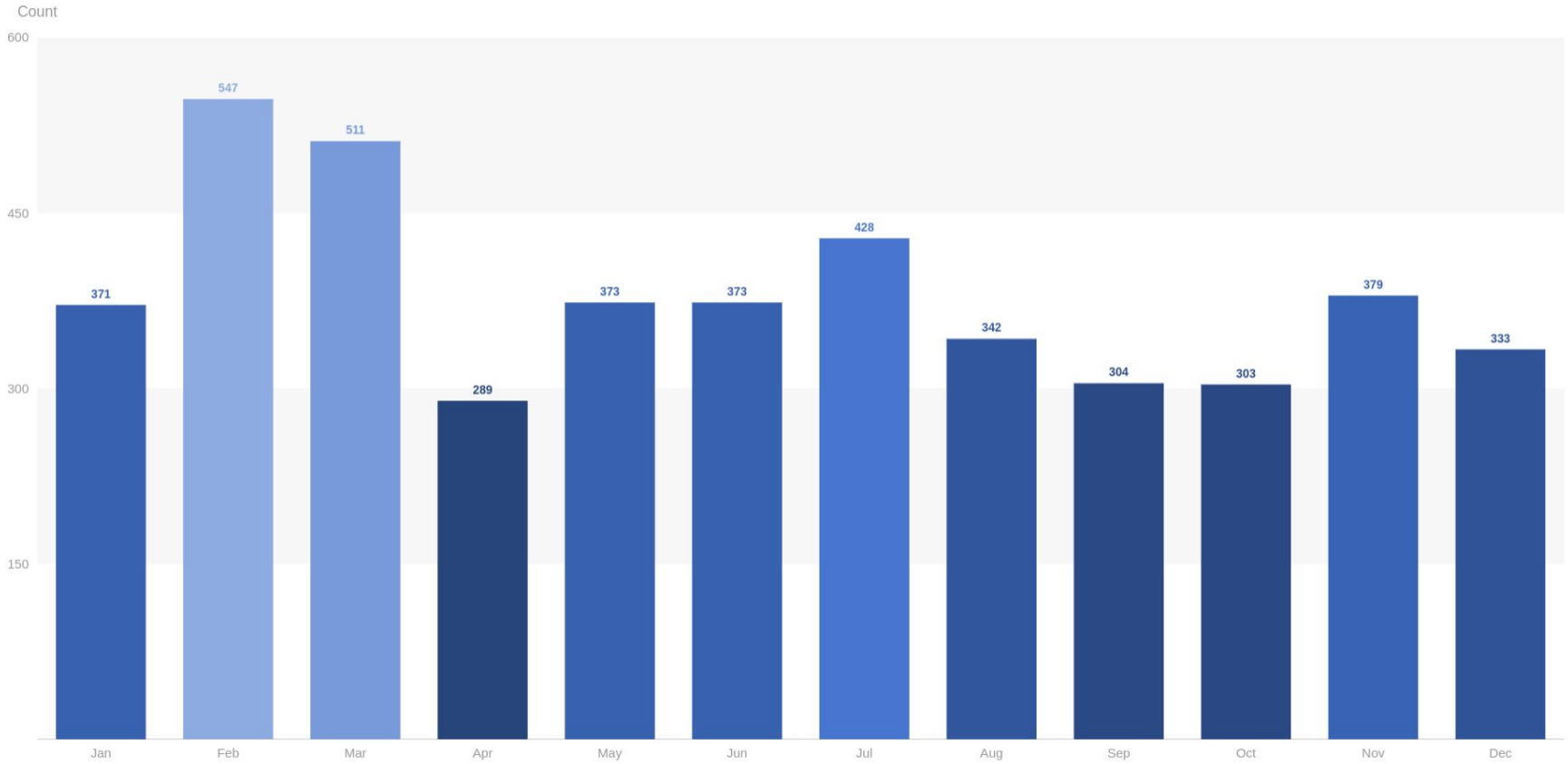
Bookmarks



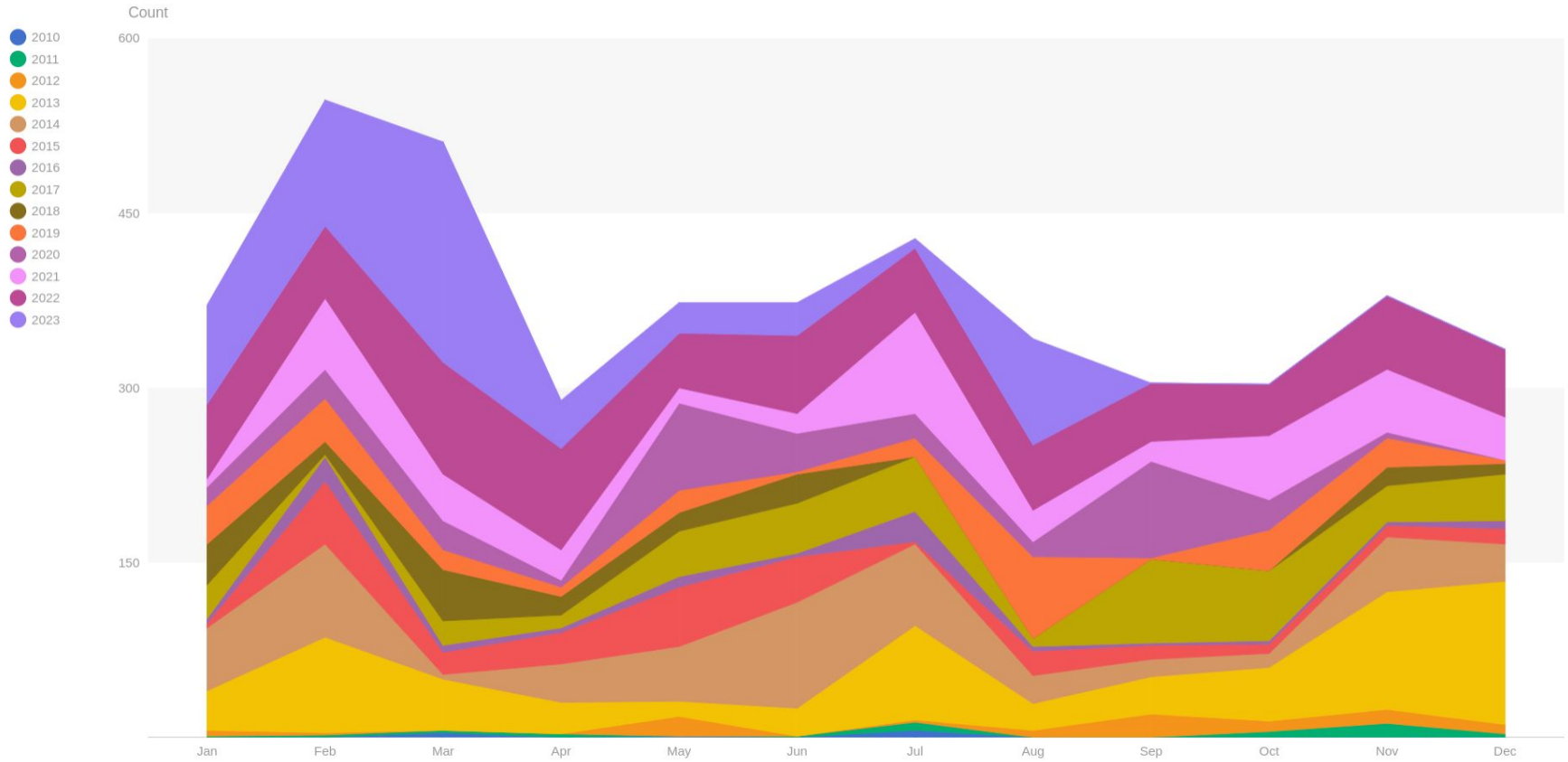
Bookmarks by year



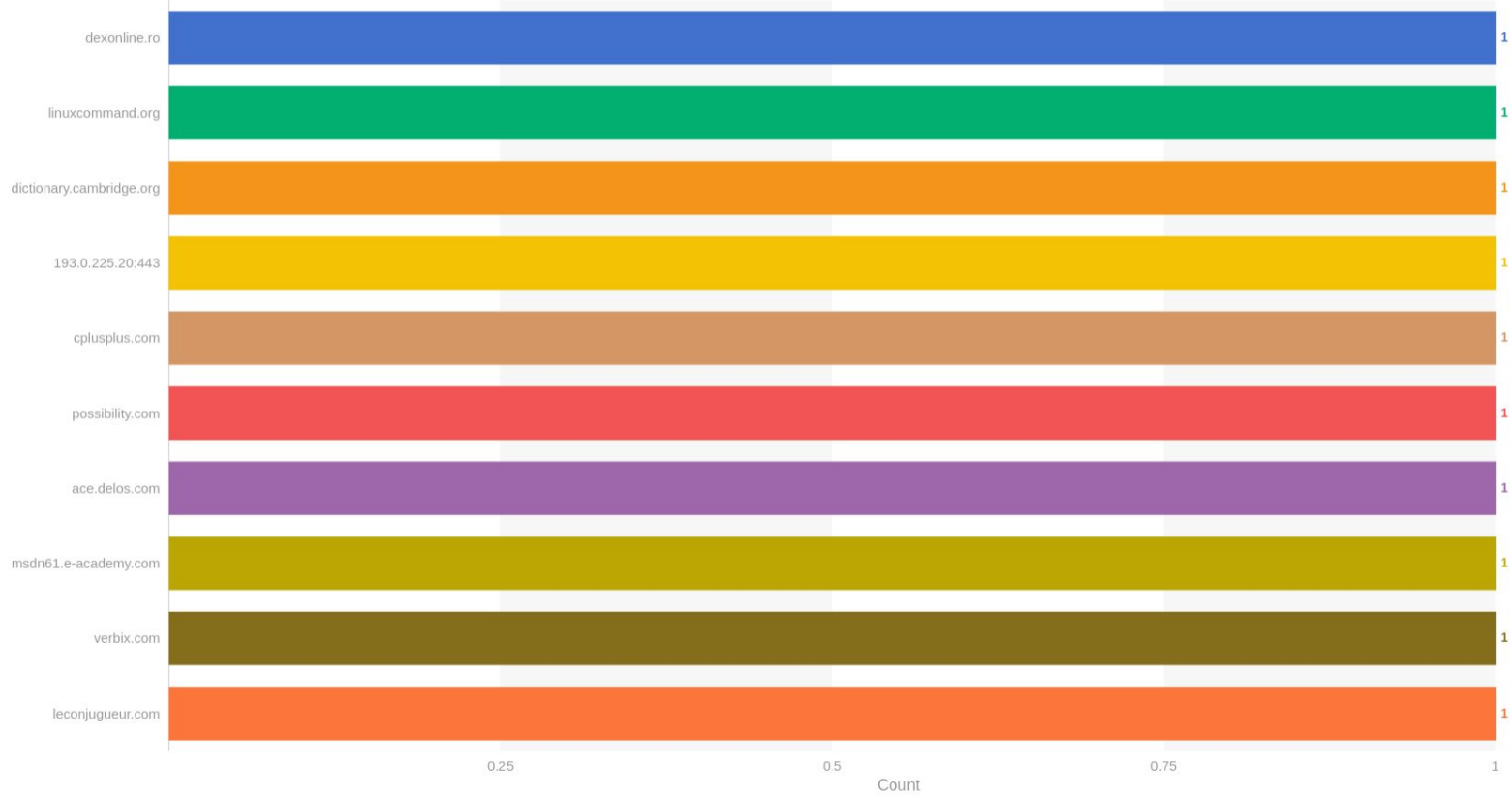
Bookmarks by month



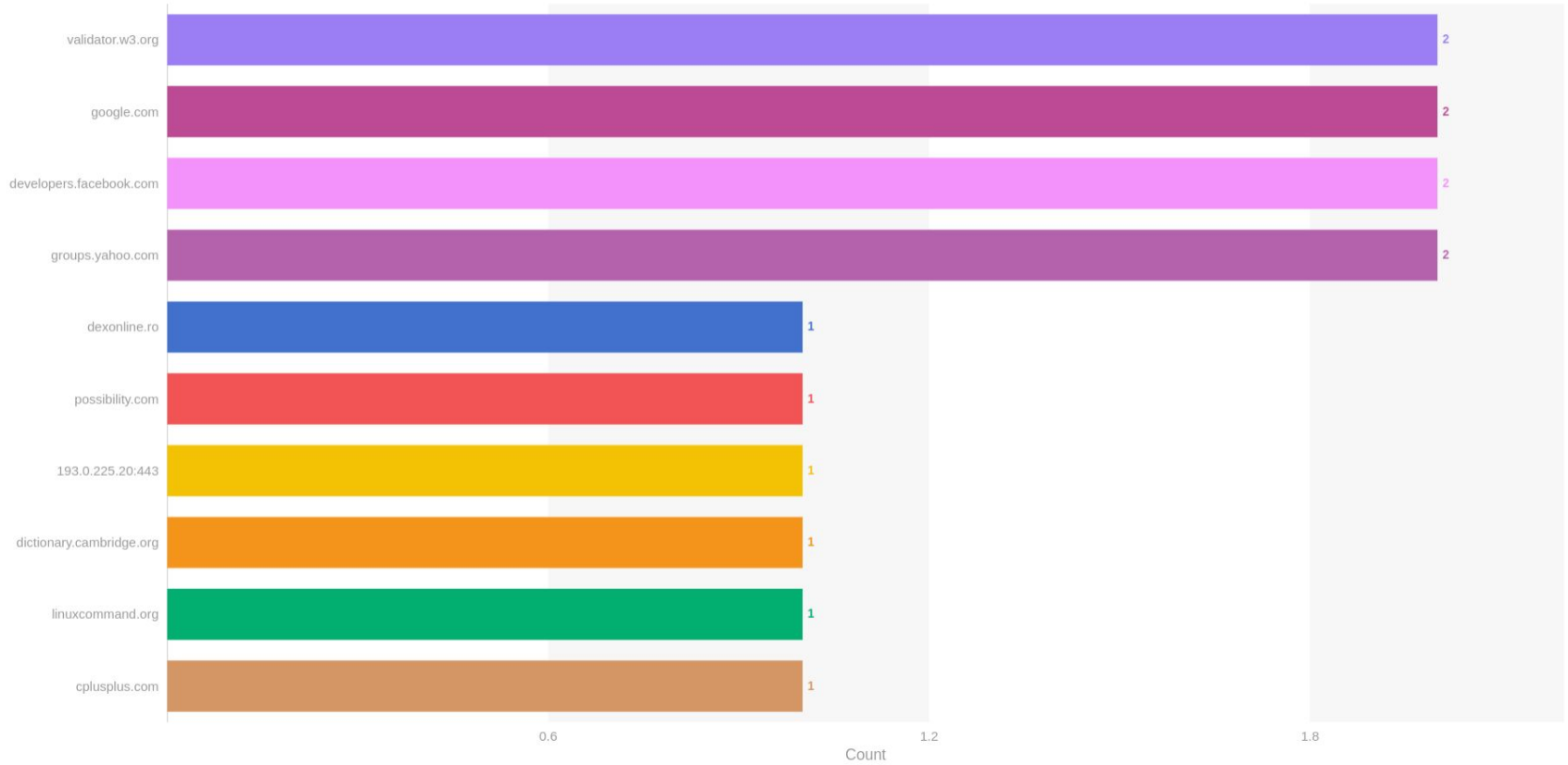
Bookmarks by month



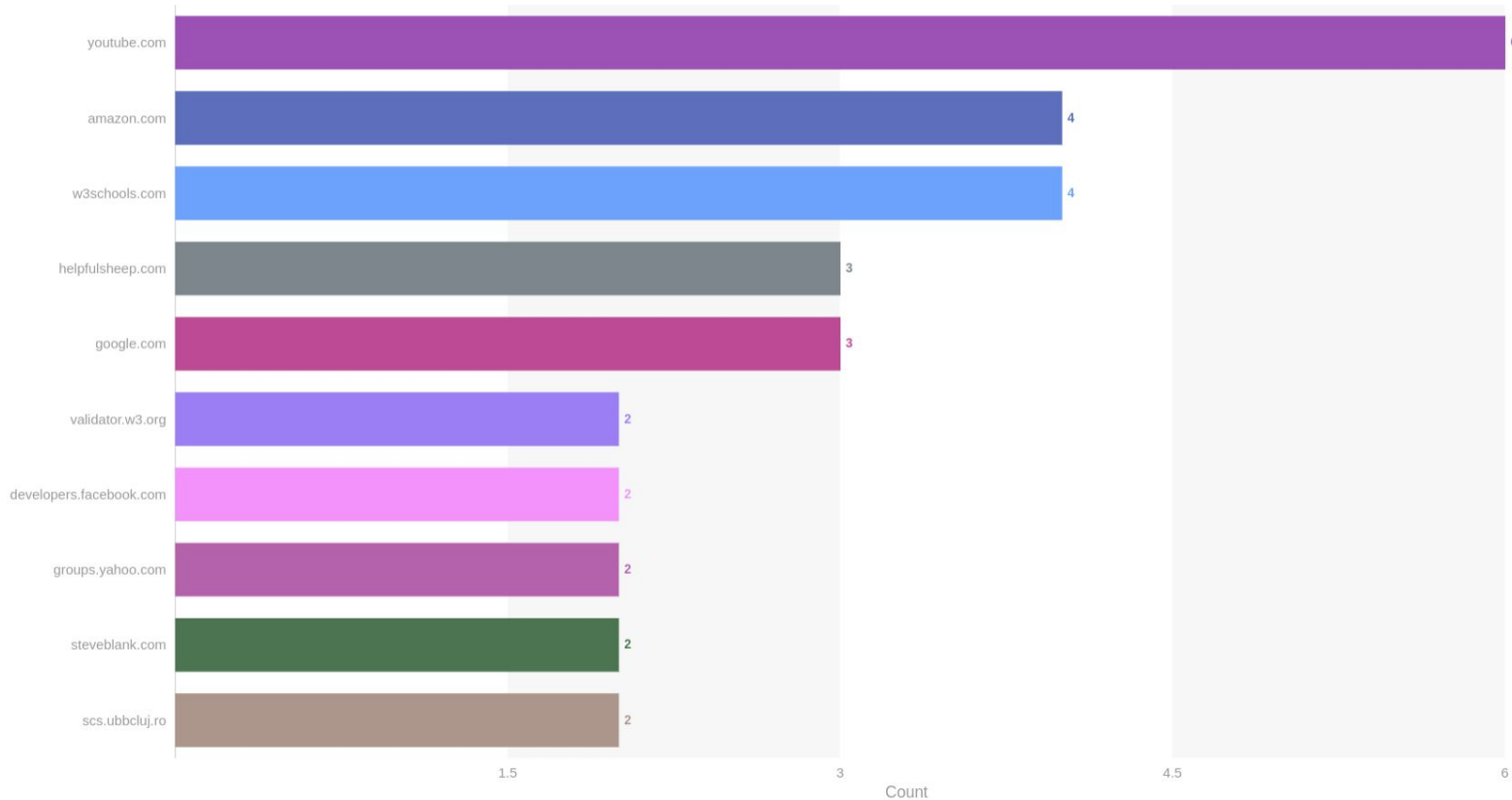
Top 10 domains 2010 - 2010



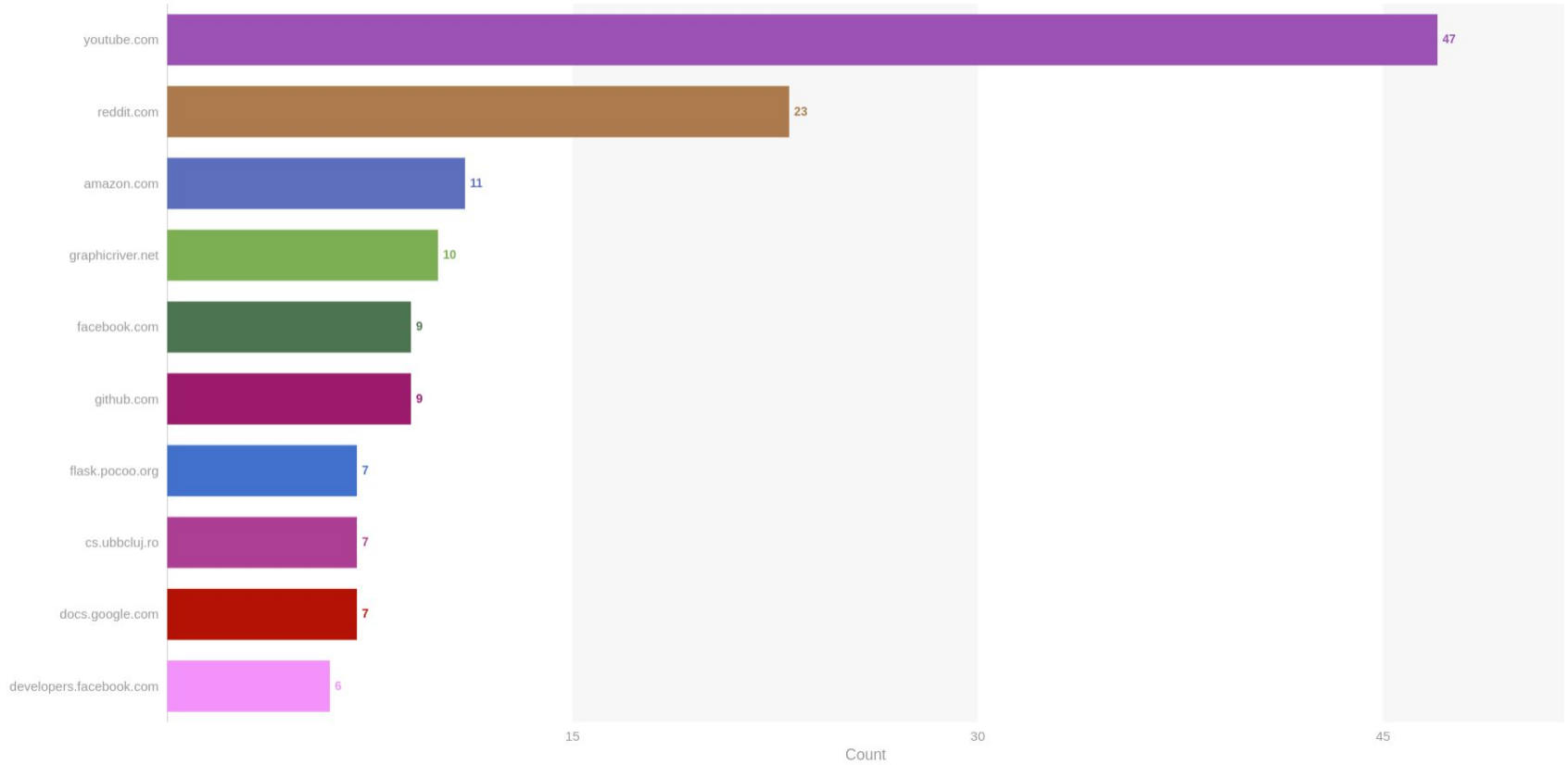
Top 10 domains 2010 - 2011



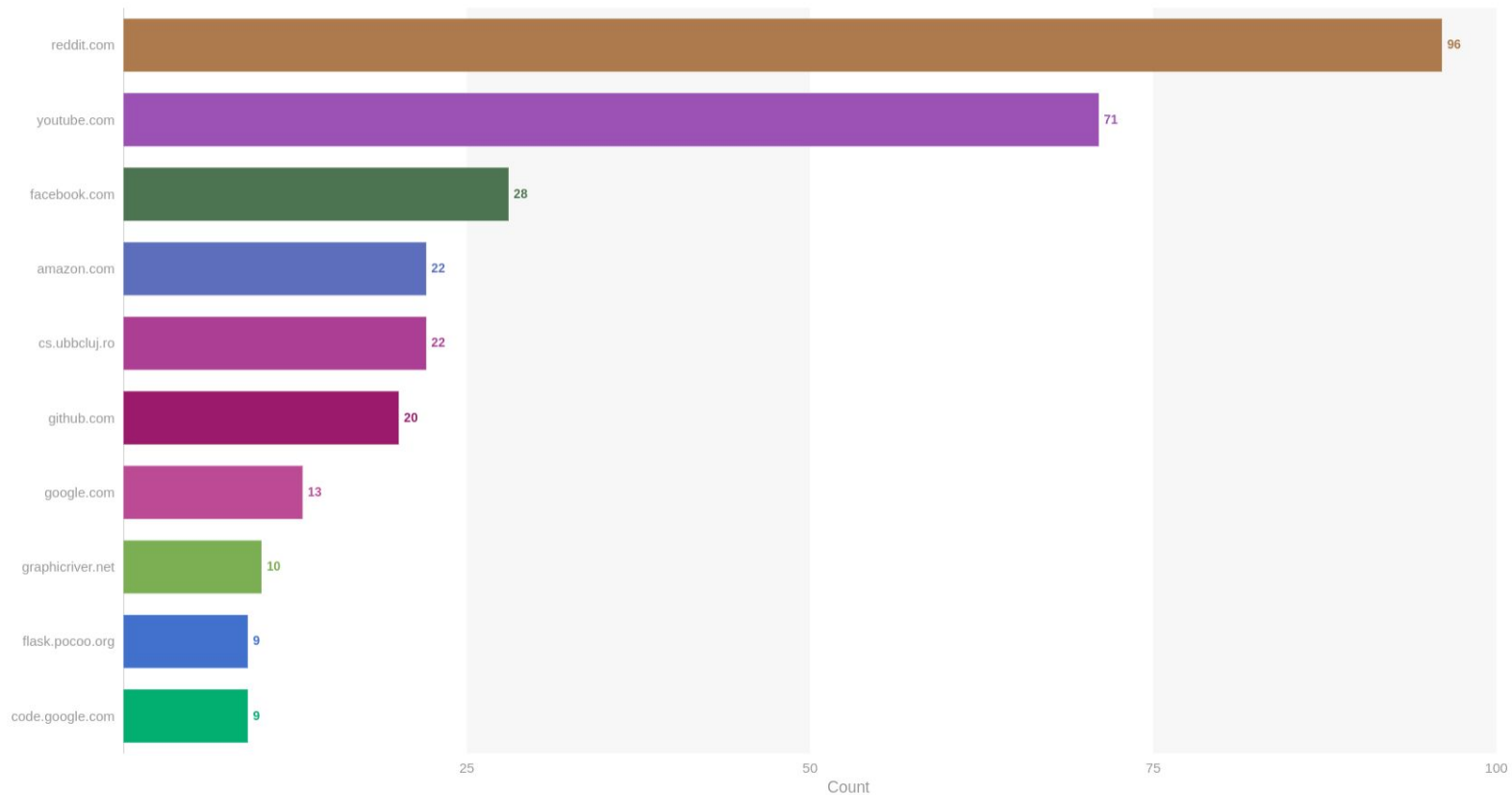
Top 10 domains 2010 - 2012



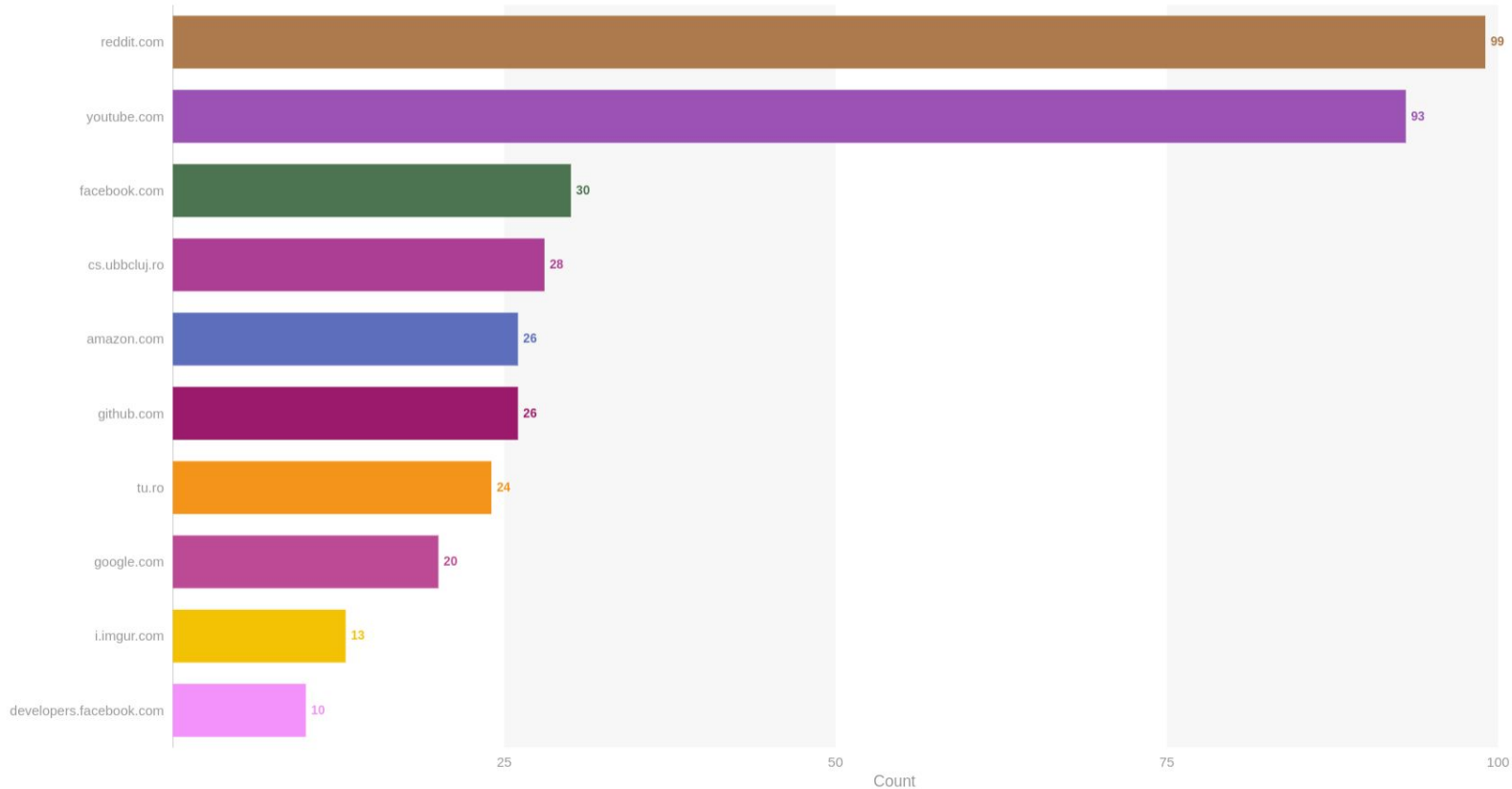
Top 10 domains 2010 - 2013



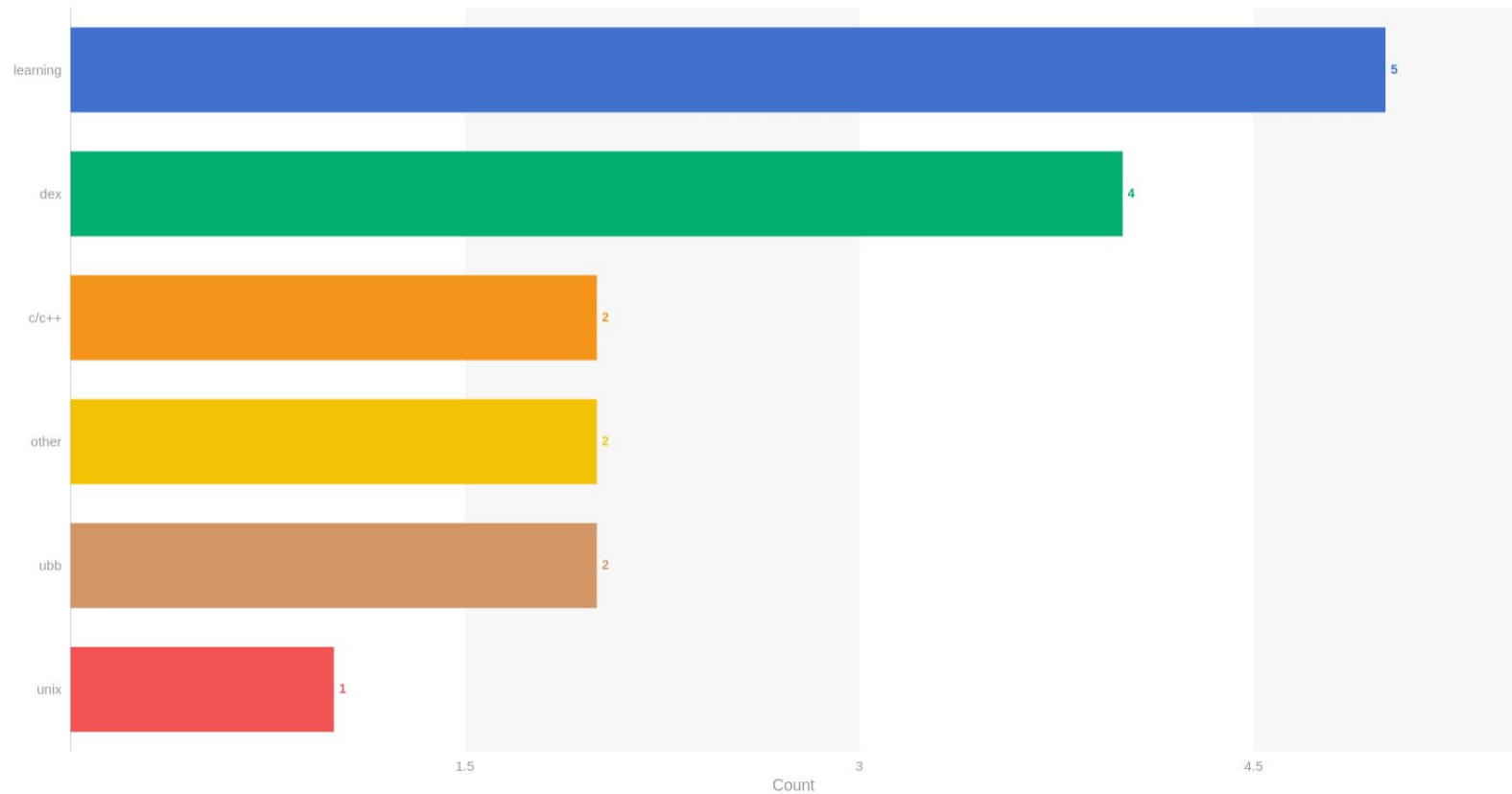
Top 10 domains 2010 - 2014



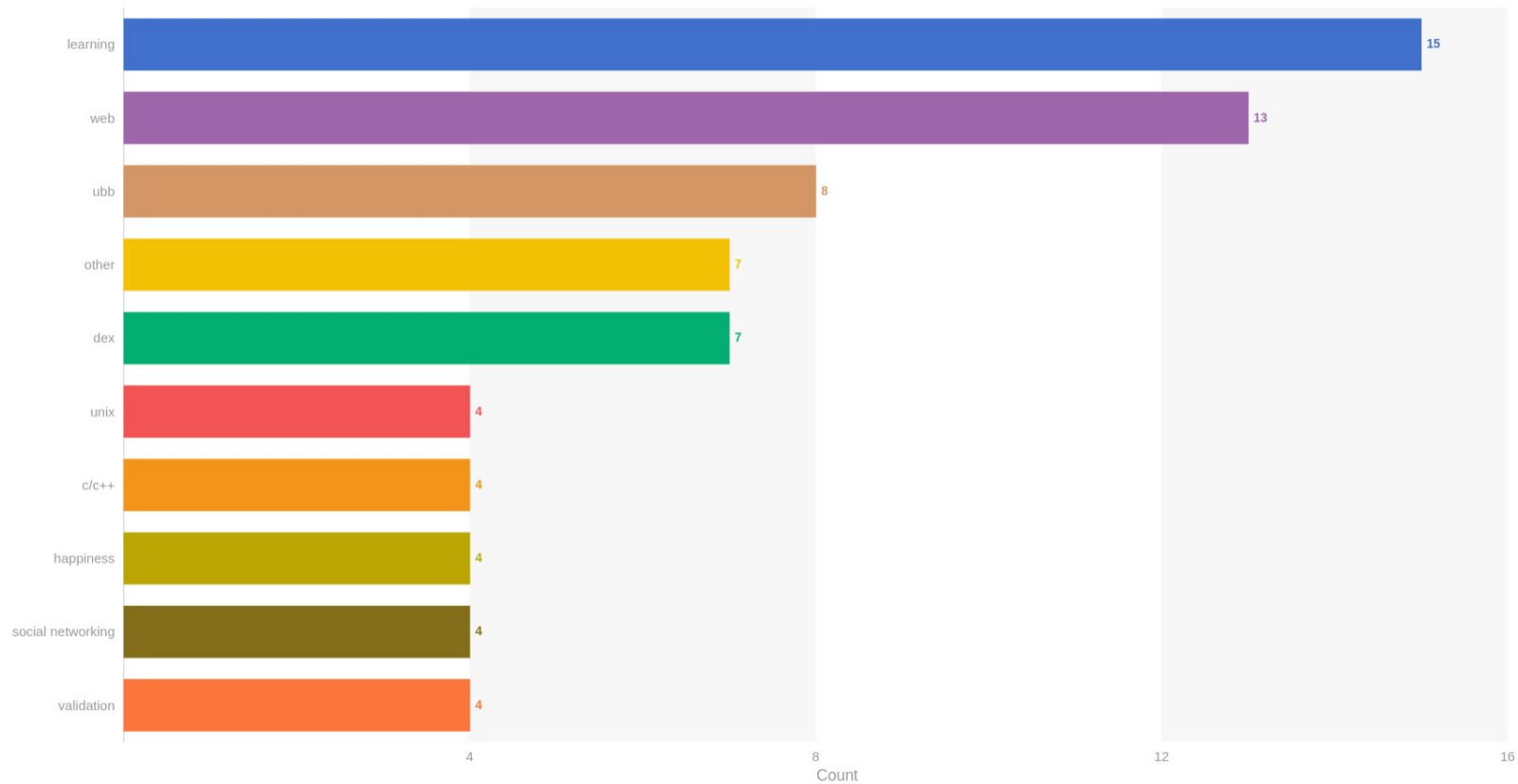
Top 10 domains 2010 - 2015



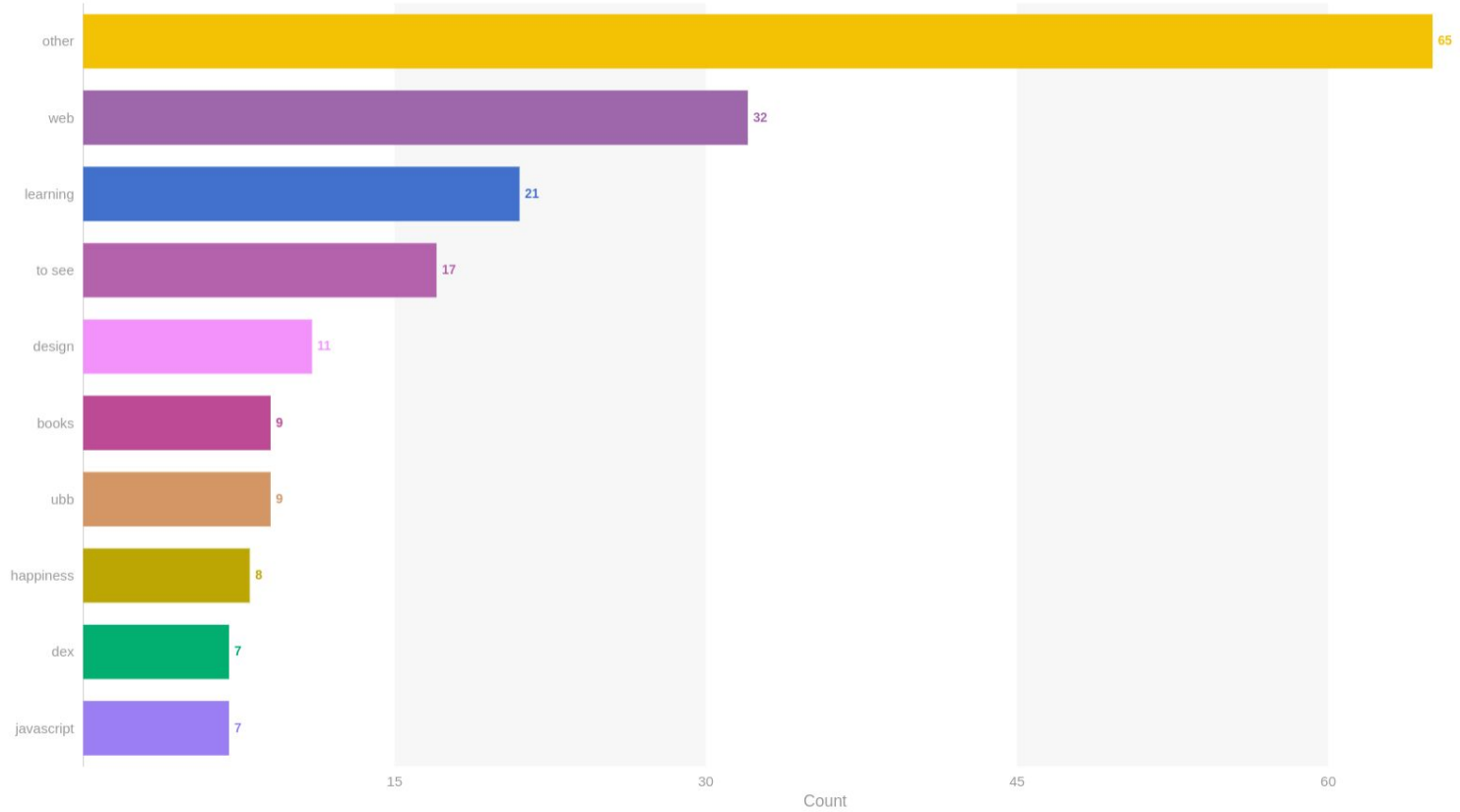
Top 10 labels 2010 - 2010



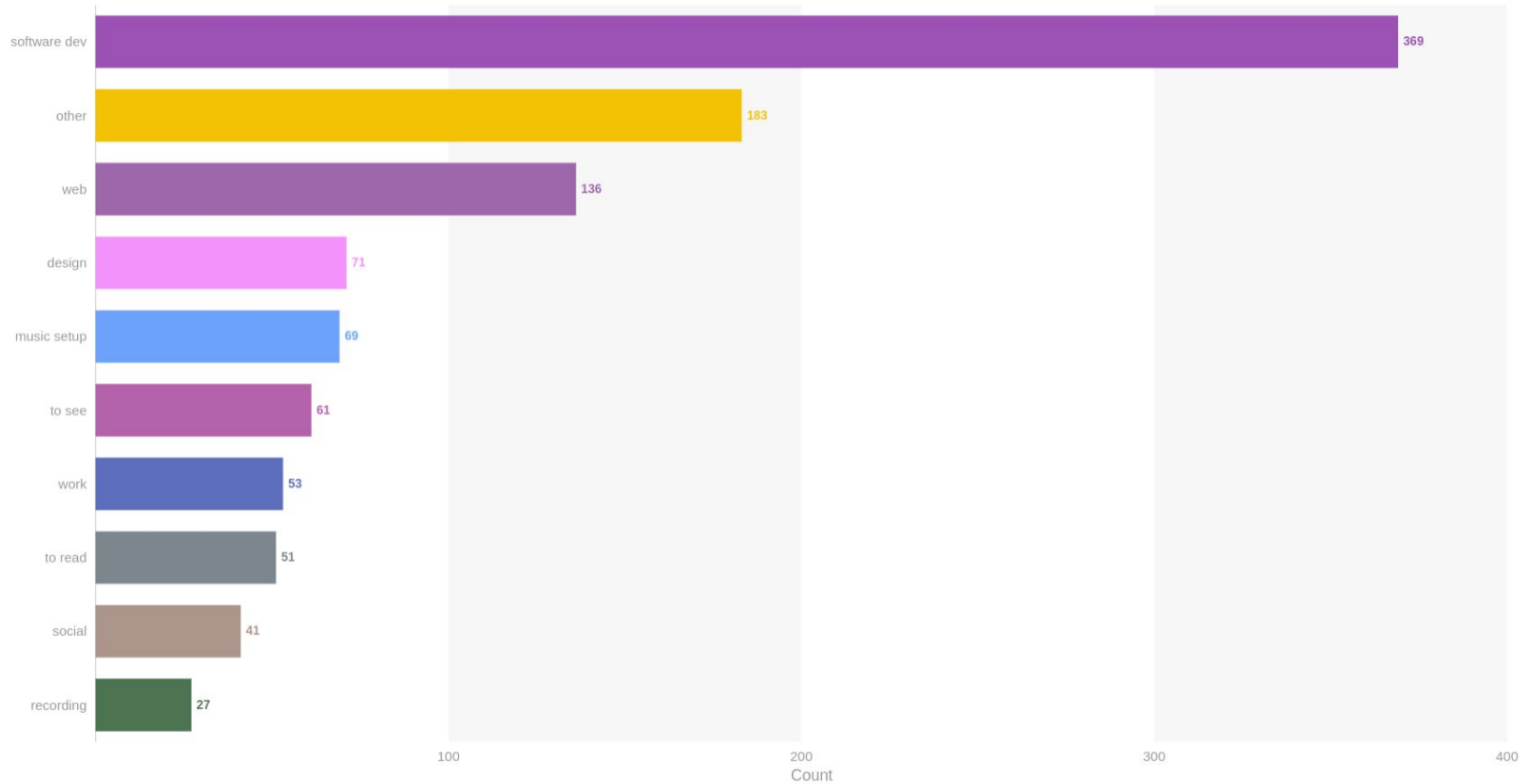
Top 10 labels 2010 - 2011



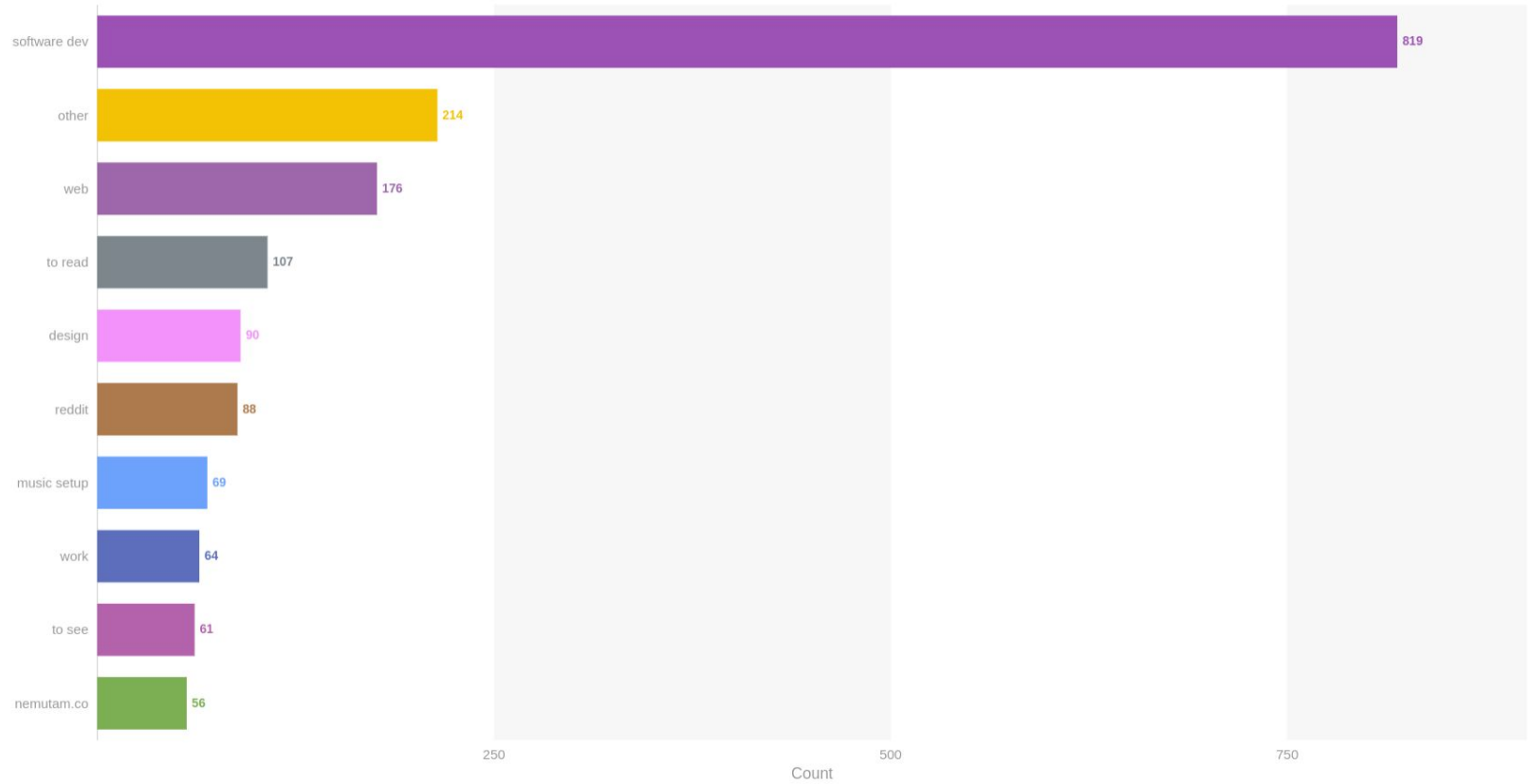
Top 10 labels 2010 - 2012



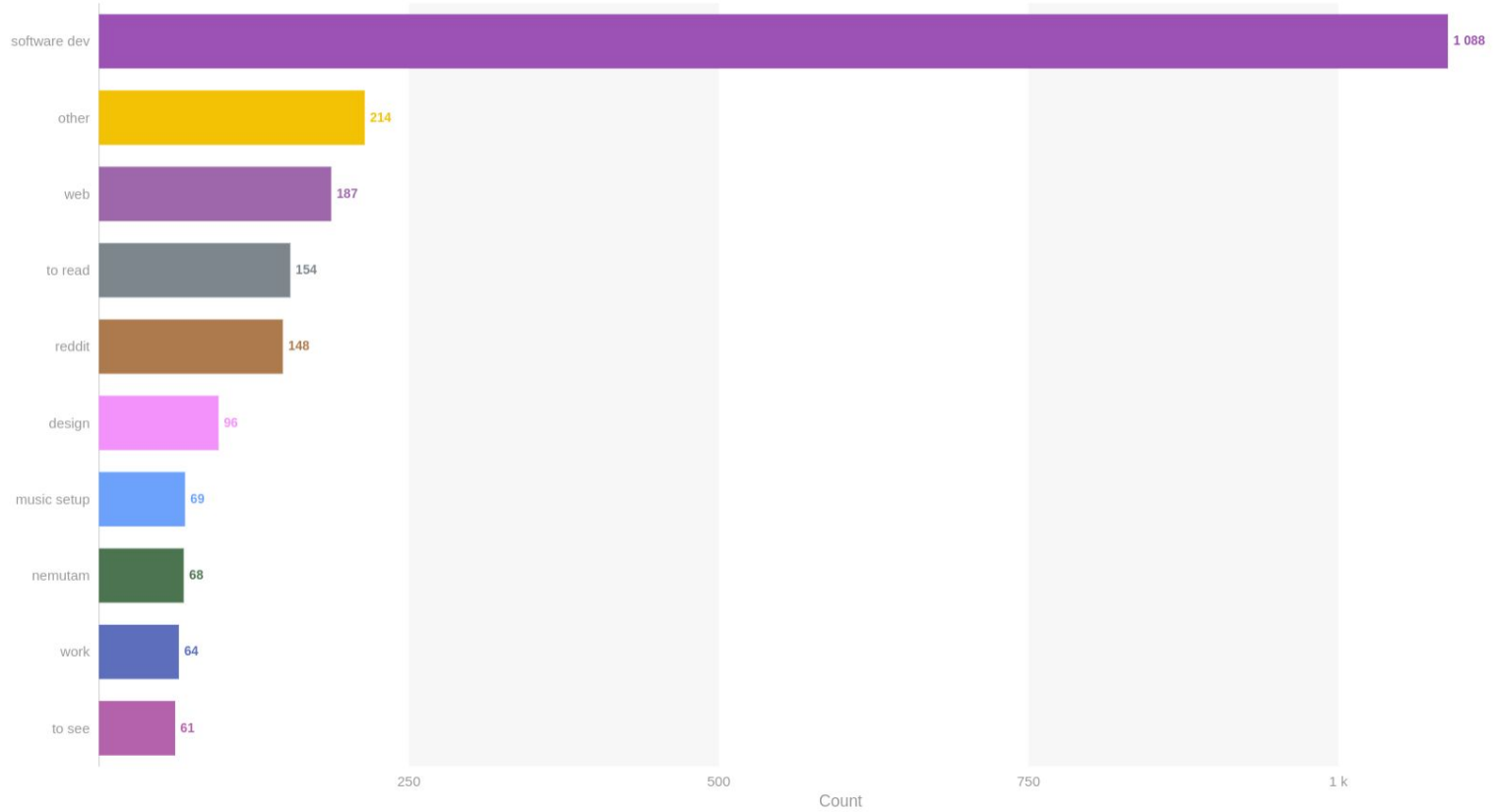
Top 10 labels 2010 - 2013



Top 10 labels 2010 - 2014



Top 10 labels 2010 - 2015





Fin!

github.com/g4b1nagy/bookmark_db

Scikit-learn and GPUs

Tim Head @betatim

I say “Arrays”, you say ...

I say “Arrays”, you say ...



I say “machine learning”, you say ...

I say “machine learning”, you say ...



A match made in heaven 🥰



NumPy

Alternatives!



But there are others 😲



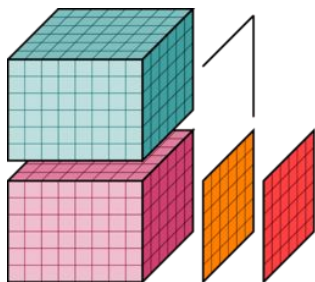
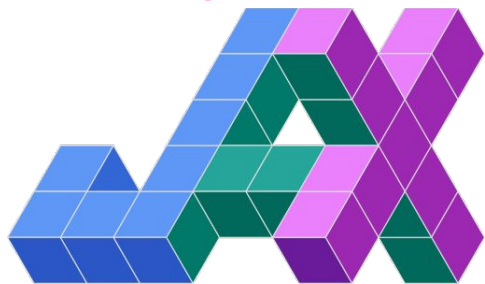
PyTorch



dask



CuPy



xarray

Unfortunately none of these work
with scikit-learn 😭😭

Until now 
(experimental support)

```
>>> X_np, y_np = make_classification(random_state=0)
>>> X_cu = cupy.asarray(X_np)
>>> y_cu = cupy.asarray(y_np)
>>> X_cu.device
<CUDA Device 0>

>>> with config_context(array_api_dispatch=True):
...     lda = LinearDiscriminantAnalysis()
...     X_trans = lda.fit_transform(X_cu, y_cu)
>>> X_trans.device
<CUDA Device 0>
```

There is a lot of work left to do! Help us?!

Join us: <https://github.com/scikit-learn/scikit-learn/issues/26024>

Visit issue **#26024** on the scikit-learn GitHub repository to help out adding GPU support to scikit-learn.

Thank you!



Vega-Altair

Declarative Visualization in Python



binder_stefan@outlook.com

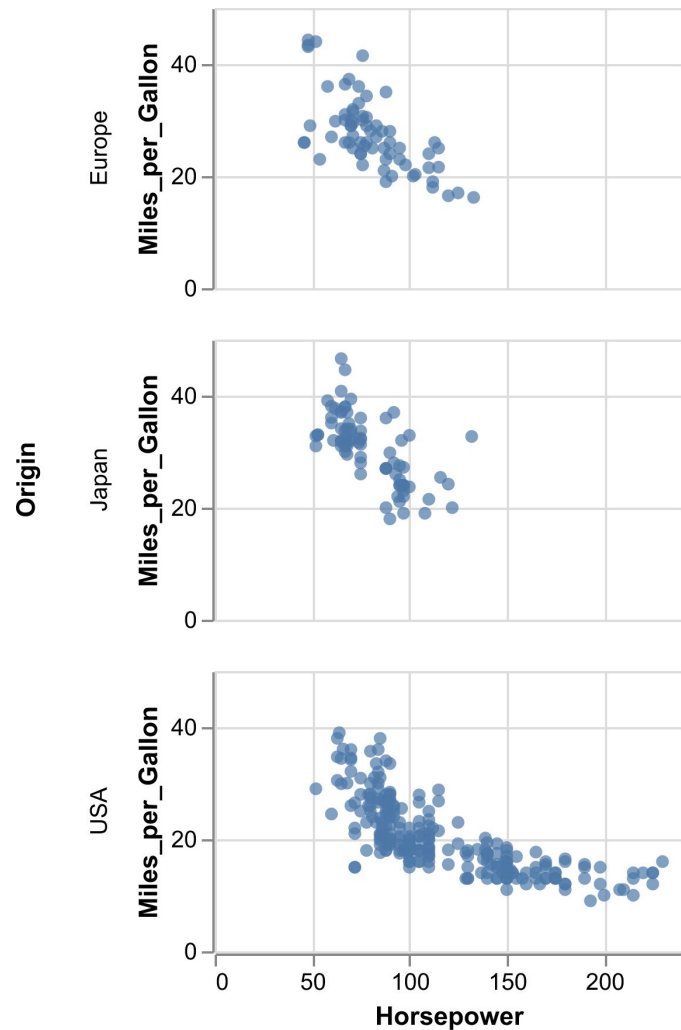


[@binste](#)

What is Altair? | Example

```
import altair as alt

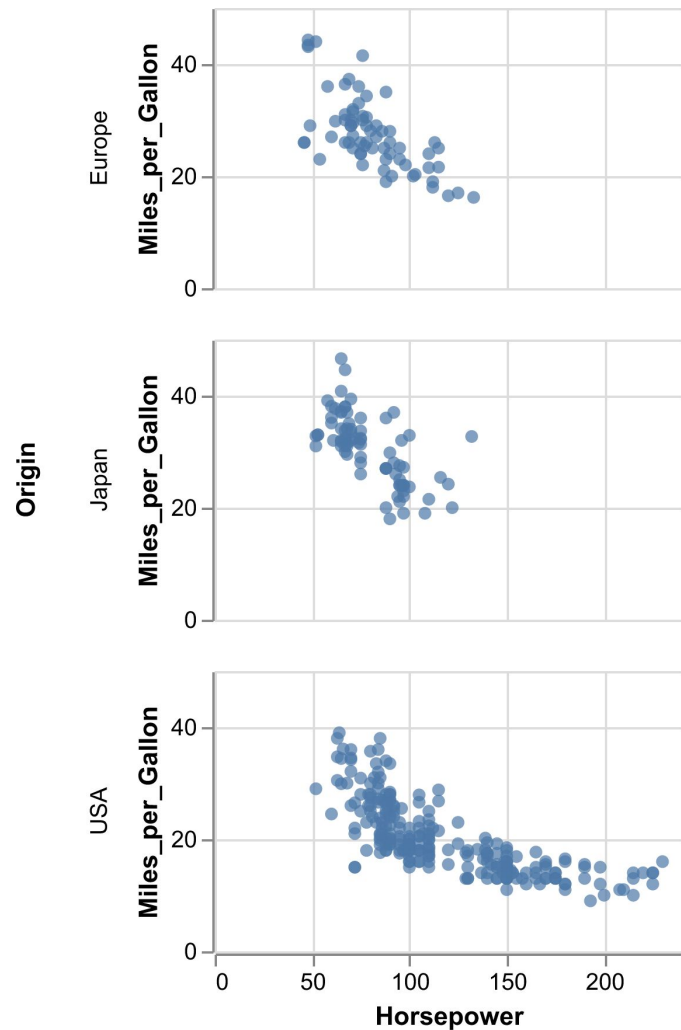
alt.Chart(cars_df).mark_circle().encode(
    alt.X("Horsepower"),
    alt.Y("Miles_per_Gallon"),
    alt.Row("Origin")
)
```



What is Altair? | Example

```
import altair as alt
```

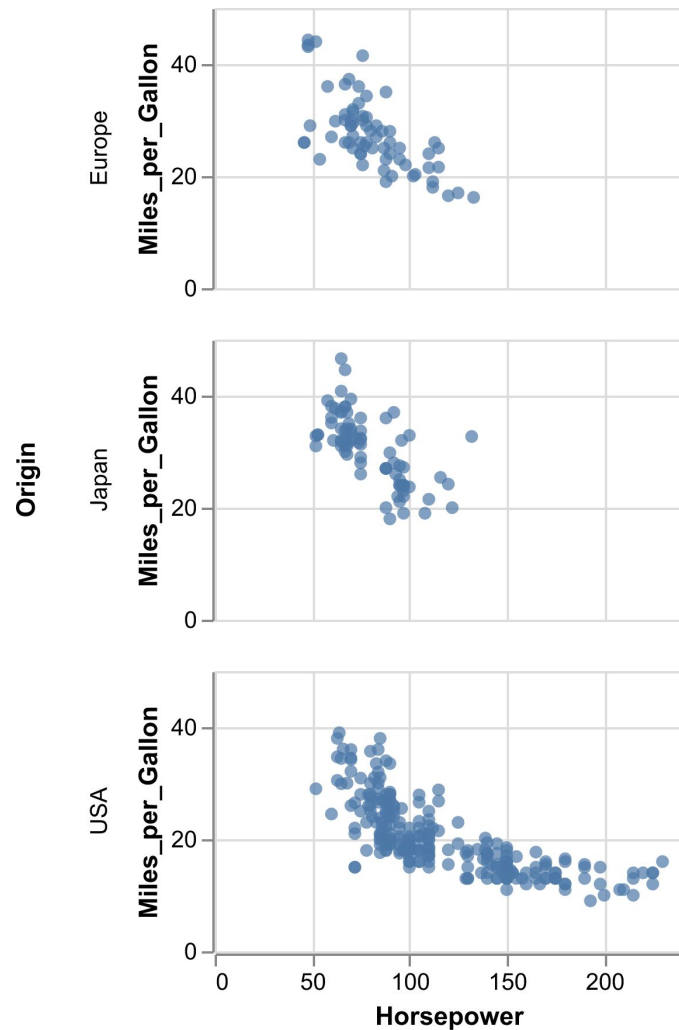
```
alt.Chart(cars_df).mark_circle().encode(  
    alt.X("Horsepower"),  
    alt.Y("Miles_per_Gallon"),  
    alt.Row("Origin")  
)
```



What is Altair? | Example

```
import altair as alt
```

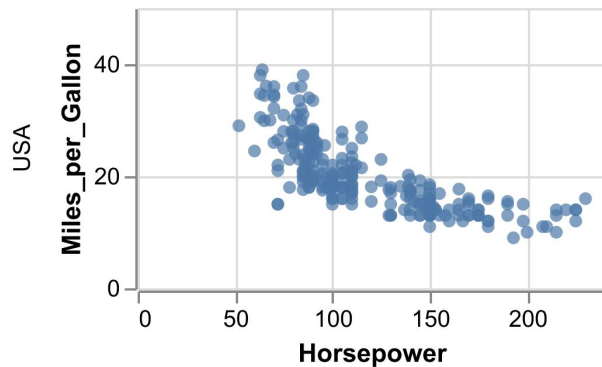
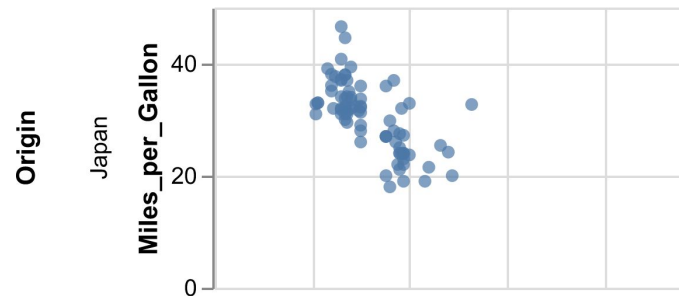
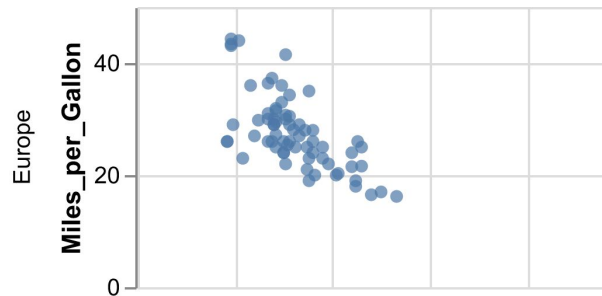
```
alt.Chart(cars_df).mark_circle().encode(  
    alt.X("Horsepower"),  
    alt.Y("Miles_per_Gallon"),  
    alt.Row("Origin")  
)
```



What is Altair? | Example

```
import altair as alt

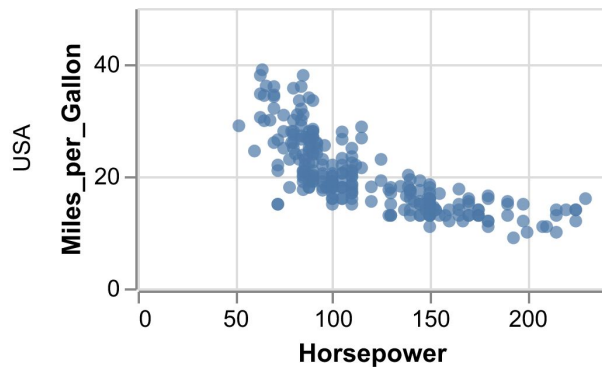
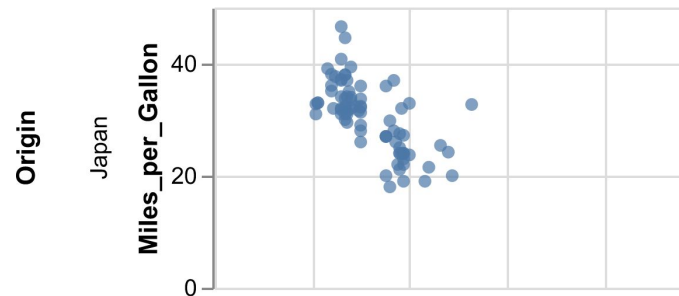
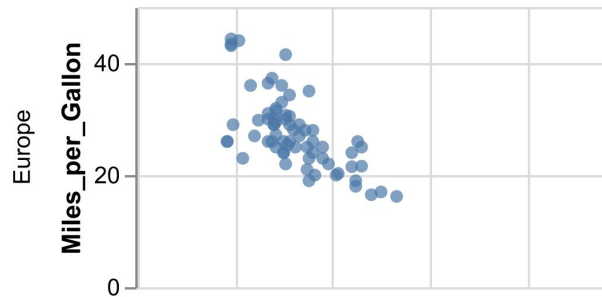
alt.Chart(cars_df).mark_circle().encode(
    alt.X("Horsepower"),
    alt.Y("Miles_per_Gallon"),
    alt.Row("Origin")
)
```



What is Altair? | Example

```
import altair as alt
```

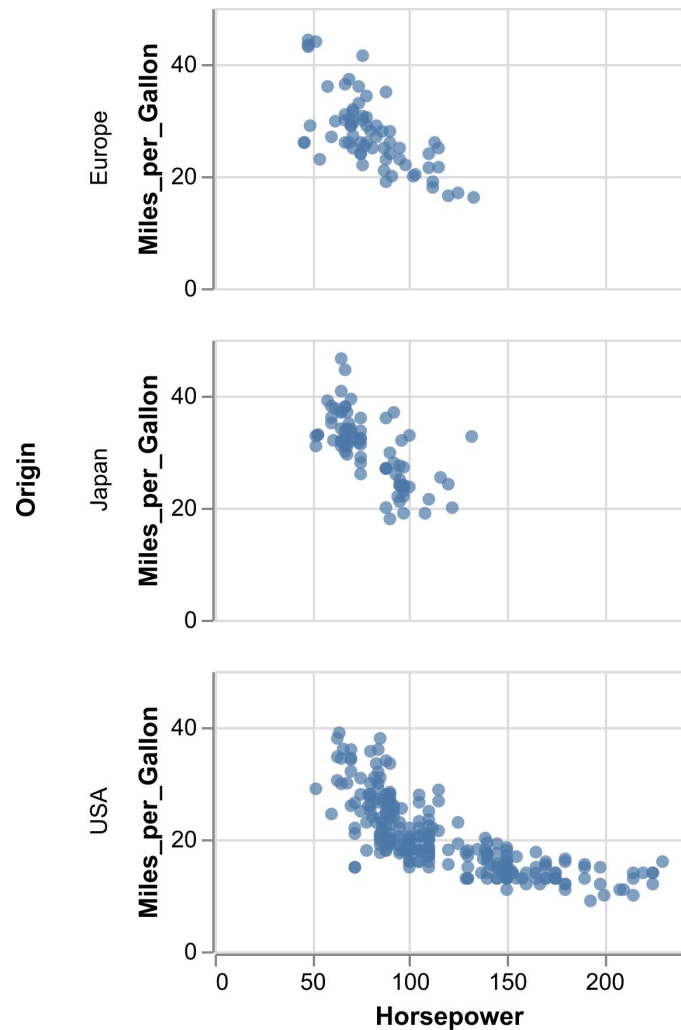
```
alt.Chart(cars_df).mark_circle().encode(  
    alt.X("Horsepower"),  
    alt.Y("Miles_per_Gallon"),  
    alt.Row("Origin")  
)
```



What is Altair? | Example

```
import altair as alt

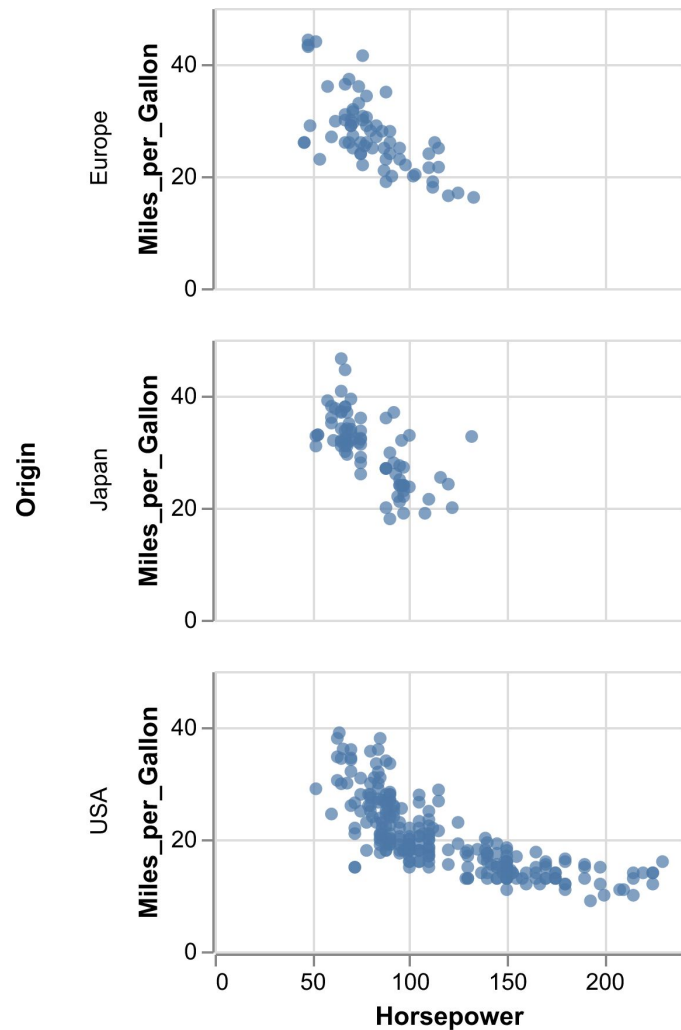
alt.Chart(cars_df).mark_circle().encode(
    alt.X("Horsepower"),
    alt.Y("Miles_per_Gallon"),
    alt.Row("Origin")
)
```



What is Altair? | Example

```
import altair as alt

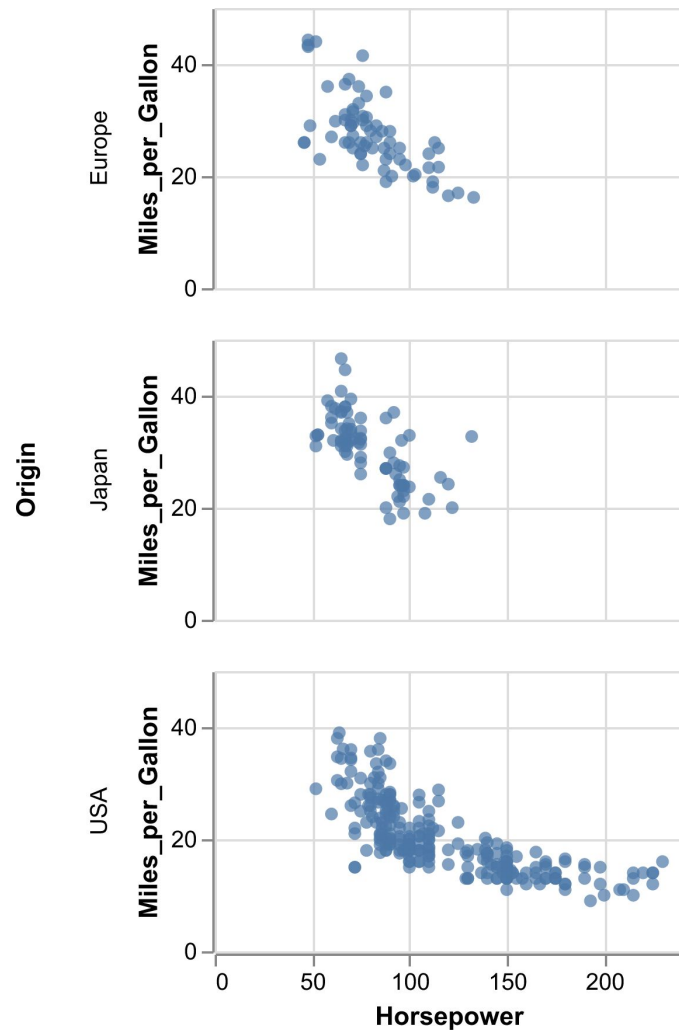
alt.Chart(cars_df).mark_circle().encode(
    alt.X("Horsepower"),
    alt.Y("Miles_per_Gallon"),
    alt.Row("Origin")
)
```



What is Altair? | Example

```
import altair as alt

alt.Chart(cars_df).mark_circle().encode(
    alt.X("Horsepower"),
    alt.Y("Miles_per_Gallon"),
    alt.Row("Origin")
)
```



What is Altair? | More reasons to love it

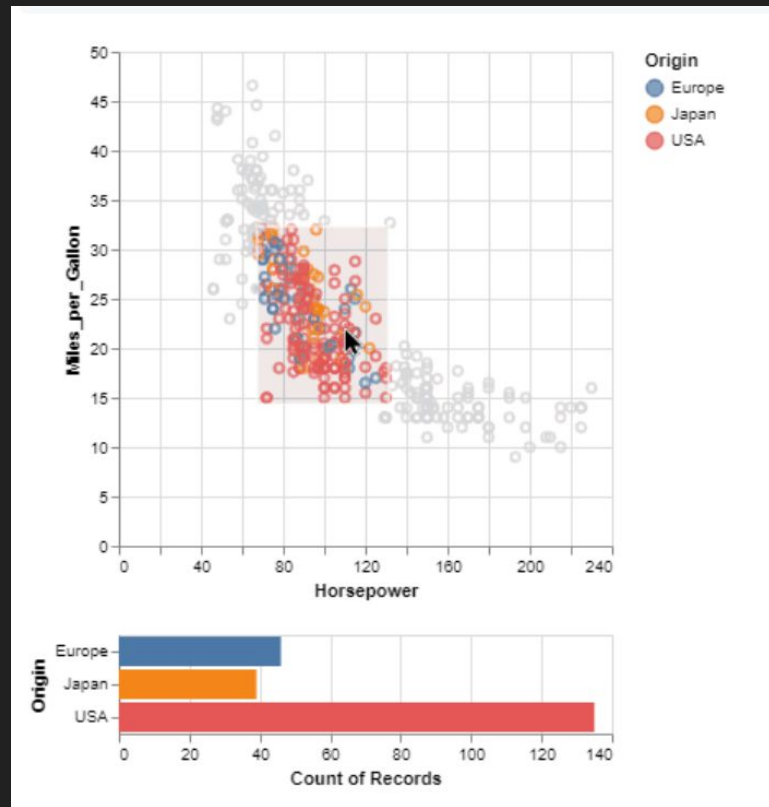
Declarative

Composable

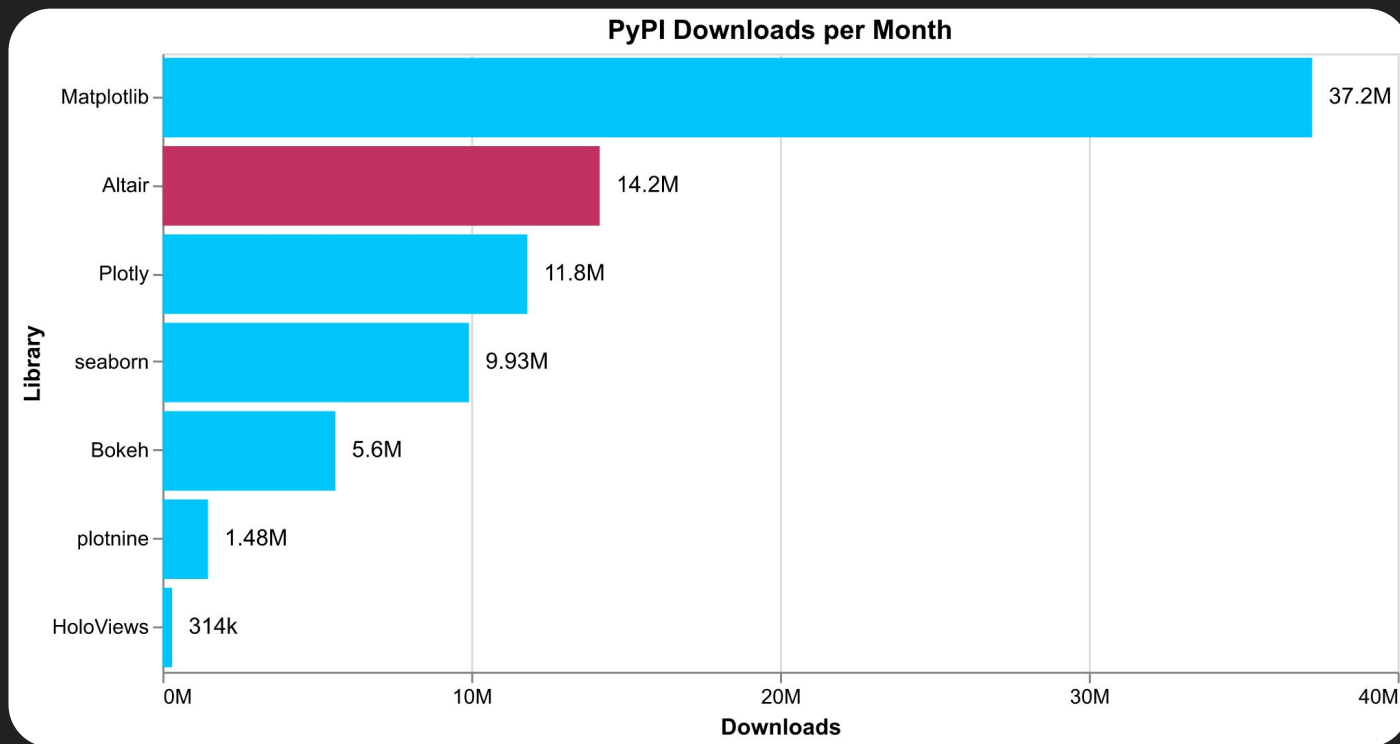
Flexible Interactivity

Popularity

...



What is Altair? | More reasons to love it



Monthly PyPI Downloads, collected with pypistats.org on Sept 8th, 2023

New developments | Altair 5

Access chart selections in Jupyter notebooks

Simplified static image export

Revamped documentation

DataFrame Interchange Protocol
(Arrow, Polars, ...)

Significantly better error messages

...



New developments | VegaFusion

Transforms data in
Python kernel

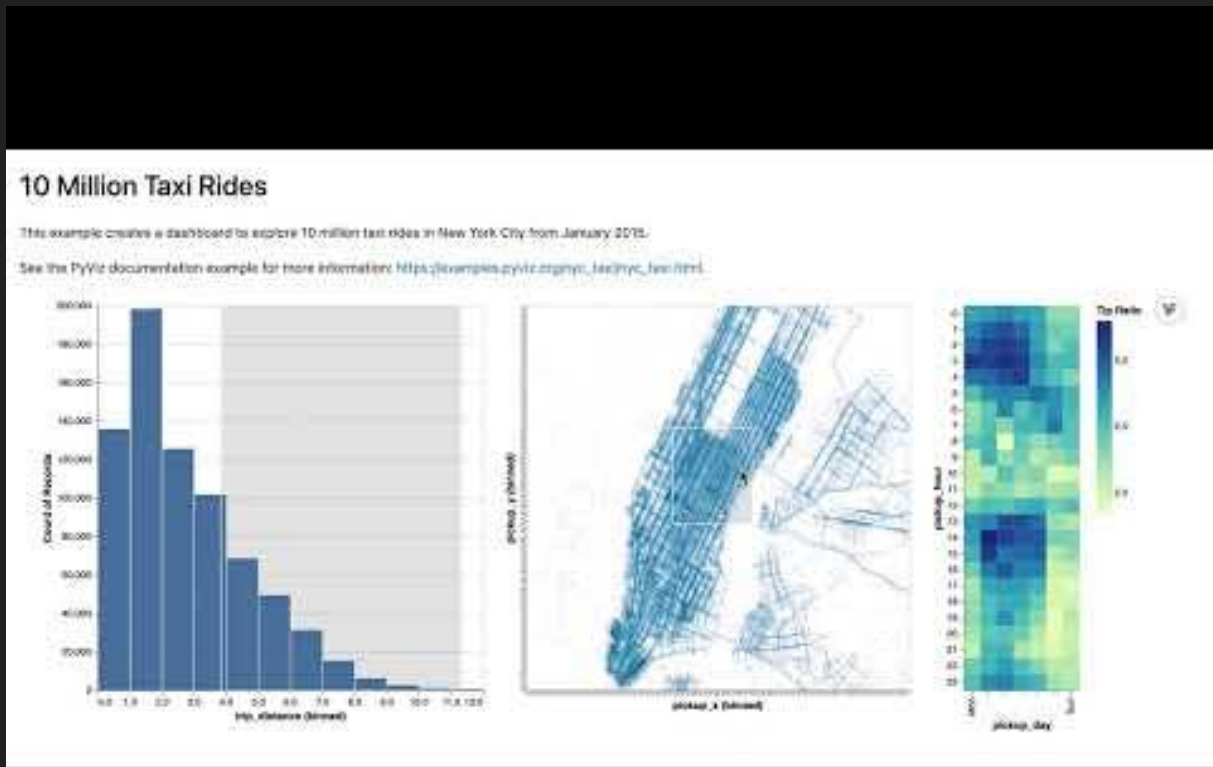
Implemented in Rust

Can push calculations
to external SQL
systems (Snowflake,
...)

Integrated in Altair 5.1



hex-inc/vegafusion



New developments | altair_tiles

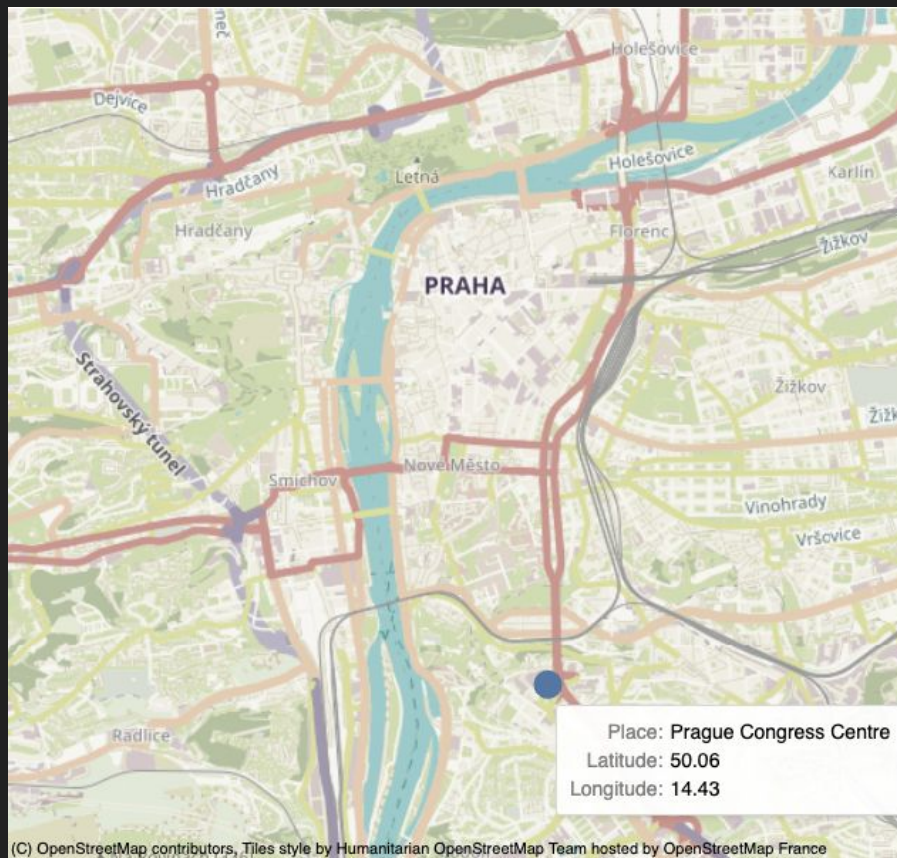
Add map tiles such as
from OpenStreetMap
to your Altair
visualizations

Early development

First release expected
“soon”



[altair-viz/altair_tiles](https://github.com/altair-viz/altair_tiles)



Learn more

Documentation: altair-viz.github.io

Code / Issues / Discussion: github.com/altair-viz/altair



binder_stefan@outlook.com



@binste