WHO PUT THE PYTHON IN MY BROWSER?! A QUICK GUIDE TO PYSCRIPT!

PRESENTED BY: SADUKIE

PyOhio 2023

WHAT IS PYSCRIPT?

- Python in WebAssembly!
  - Pyodide
  - MicroPython
- Python in the browser
- Python with JavaScript
- Python + HTML!

PyOhio 2023

Read CSV with Pandas using PyScript

Playing with PyScript and the Ramen Ratings data from Kaggle

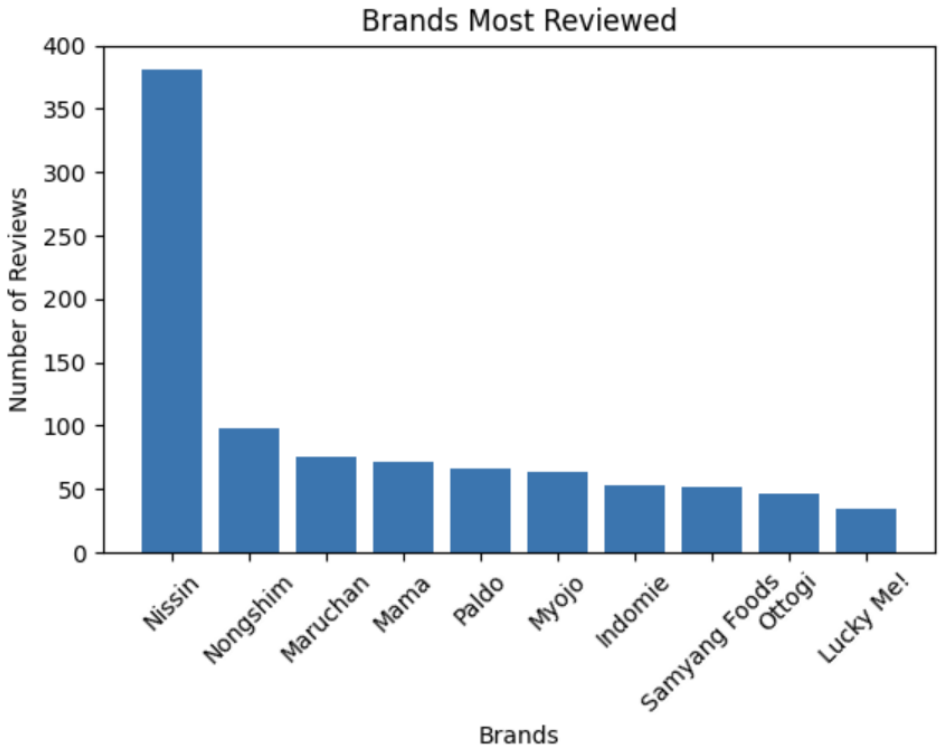
This demo uses matplotlib, pandas, and seaborn.

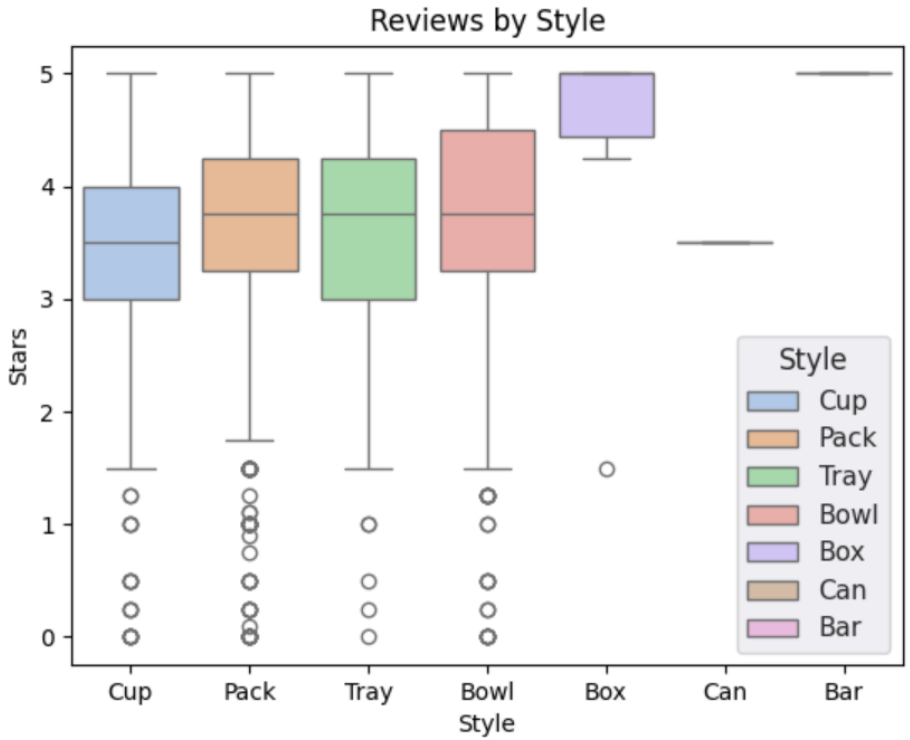
Ramen Ratings Demos

Display Parts

Load Brands Most Tested List | Load Brands Most Tested Graph | Load Seaborn Plot

	Brand
Nissin	381
Nongshim	98
Maruchan	76
Mama	71
Paldo	66
Myojo	63
Indomie	53
Samyang Foods	52
Ottogi	46
Lucky Me!	34

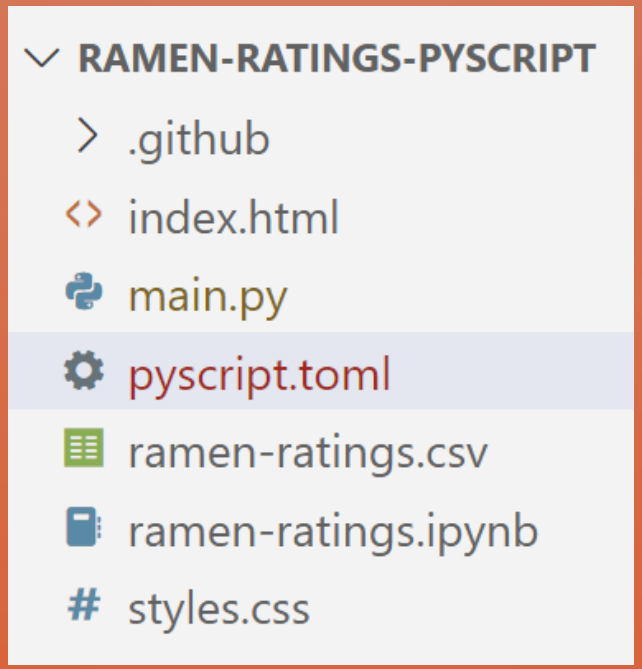




Demo: https://pyscript-demo.sadukie.com

PyOhio 2023

FILE STRUCTURE



PyOhio 2023

CONFIGURATION

TOML

- Python preferred

JSON

- Web developer preferred

TOML

```
"packages" = ["matplotlib", "pandas","seaborn"]
```

JSON

```
{
      "packages": ["matplotlib", "pandas","seaborn"]
}
```

```
<head>
    <title>PyOhio 2023: PyScript Demos</title>
    <link rel="stylesheet" href="styles.css" />
    <script type="module" src="https://pyscript.net/releases/2023.11.1/core.js"></script>
</head>
```

WHAT IS PYODIDE?

- Official site: https://pyodide.org/
- Port of CPython to WebAssembly/emscripten – compiler toolchain to WebAssembly that uses the LLVM compiler framework
- Packages installed with micropip
  - Pure Python wheels on PyPI are supported.
  - Many other packages also ported – including numpy, pandas, scipy, matplotlib, scikit-learn
- Has a REPL for testing its limits

```
<body>
    <script type="py" src="./main.py" config="./pyscript.toml"></script>
    ...
</body>
```

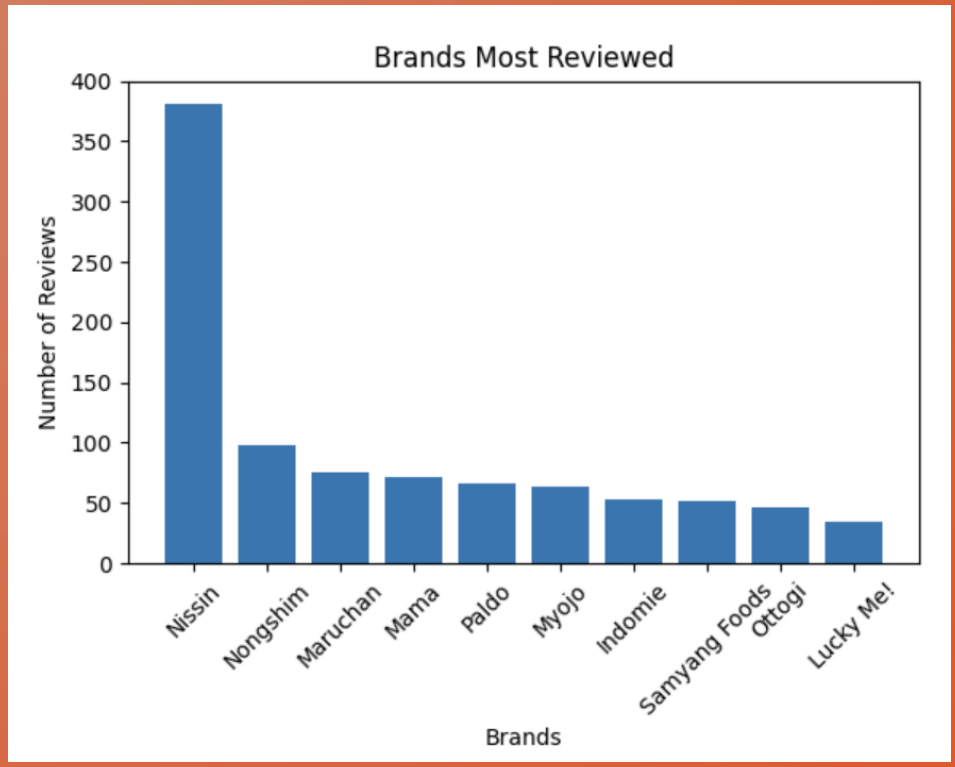
```
<body>
    <script type="py" src="./main.py" config="./pyscript.toml"></script>
    ...
            <button py-click="show_most_tested_list">Load Brands Most Tested
List</button>
            <button py-click="show_most_tested_graph">Load Brands Most Tested
Graph</button>
            <button py-click="show_most_tested_seaborn">Load Seaborn
Plot</button>
        ...
        <article>
            <section id="most_tested_list"></section>
            <section id="most_tested_graph"></section>
            <section id="most_tested_seaborn"></section>
        </article>
    ...
</body>
```

```
<body>
    <script type="py" src="./main.py" config="./pyscript.toml"></script>
    ...
            <button py-click="show_most_tested_list">Load Brands Most
Tested List</button>
        ...
         <article>
            <section id="most_tested_list"></section>
            ..
     </article>
    ...
</body>
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from pyodide.http import open_url
from pyscript import document, display


def show_most_tested_list(event):
    url_content = open_url("https://CSV_URL_HERE")
    output = document.querySelector("#most_tested_list")
    df = pd.read_csv(url_content)
    most_tested = df.Brand.value_counts().nlargest(10).to_frame()
    output.innerHTML = most_tested.to_html()
```

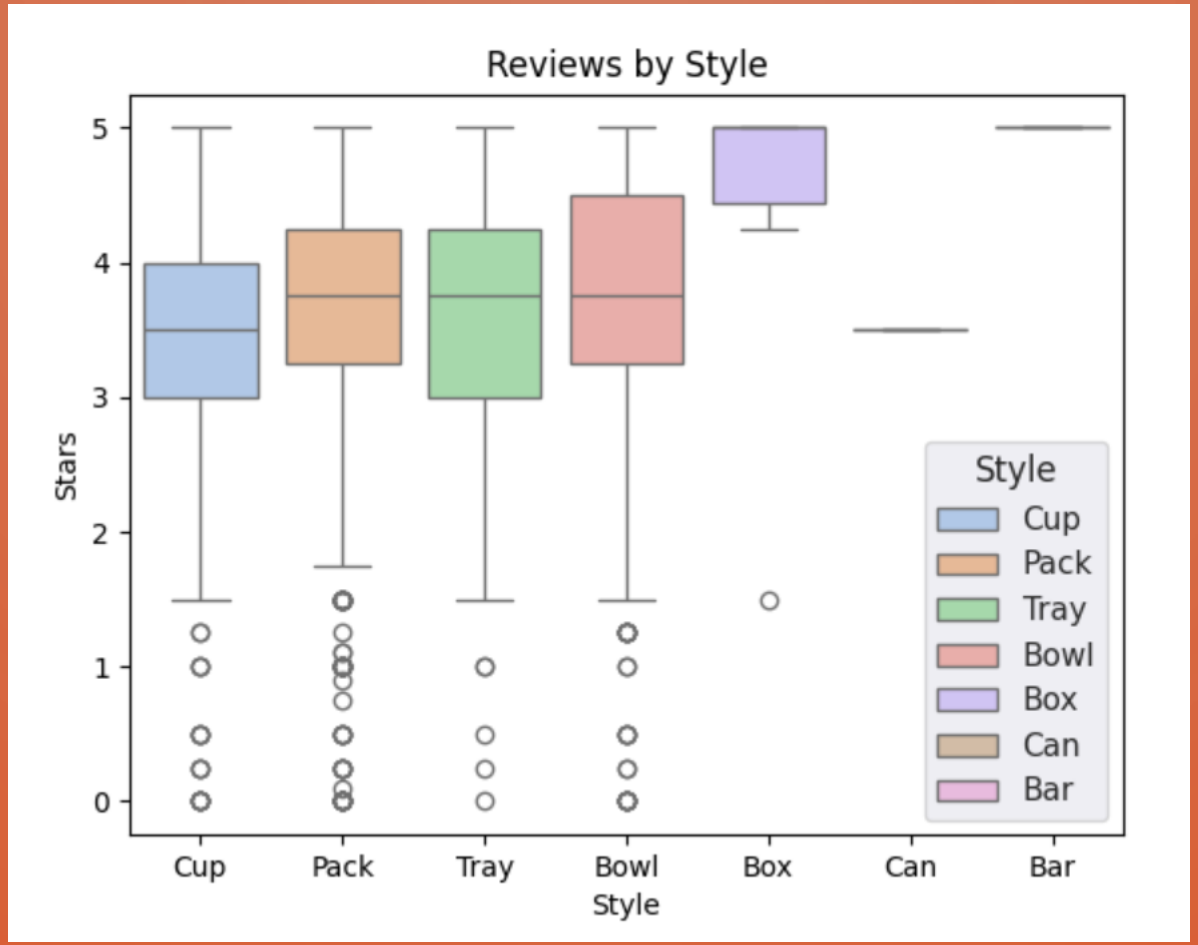
	Brand
Nissin	381
Nongshim	98
Maruchan	76
Mama	71
Paldo	66
Myojo	63
Indomie	53
Samyang Foods	52
Ottogi	46
Lucky Me!	34

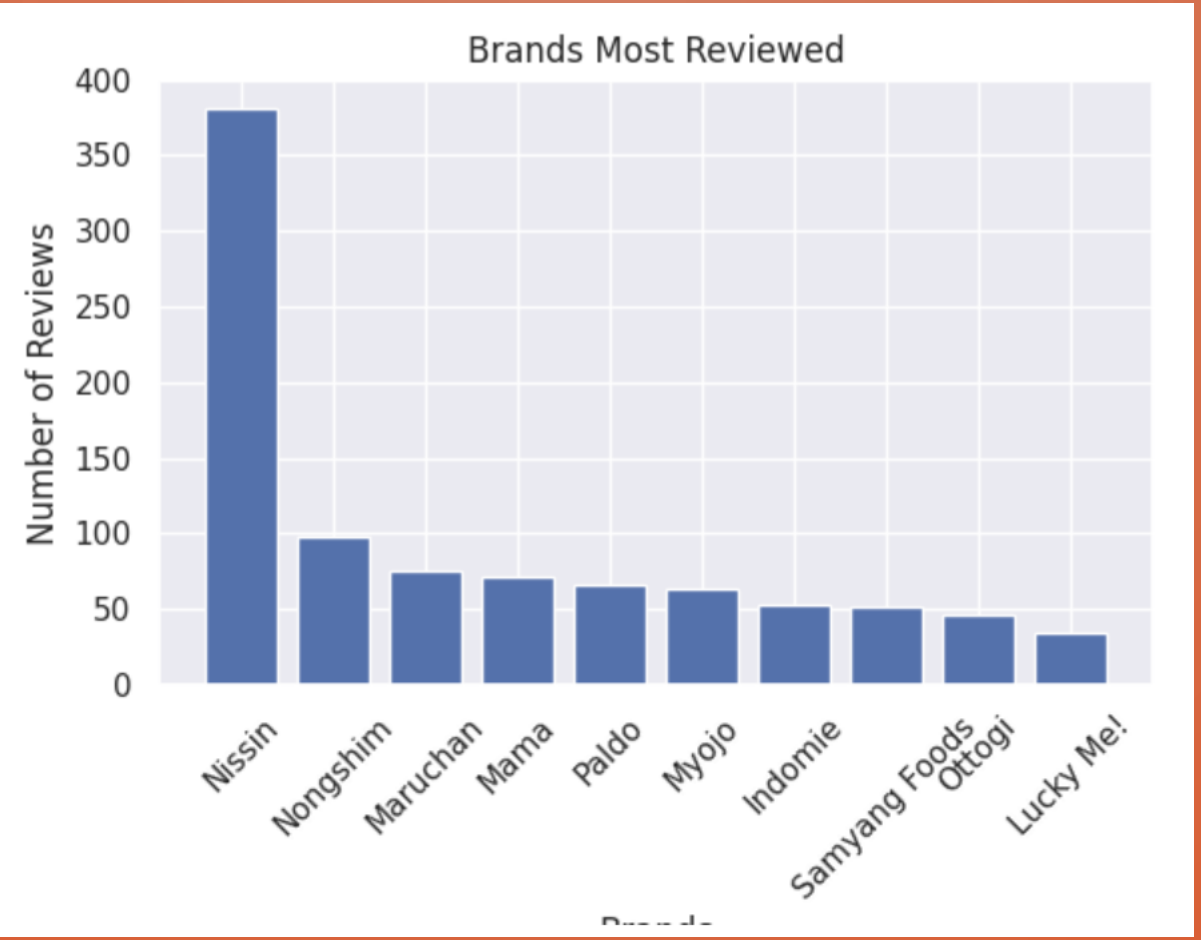
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from pyodide.http import open_url
from pyscript import document, display

def show_most_tested_graph(event):
    url_content = open_url("https://CSV_URL_HERE")
    df = pd.read_csv(url_content)
    most_tested = df.Brand.value_counts().nlargest(10).to_frame()
    fig, ax = plt.subplots()
    ax.bar(most_tested.index,most_tested.Brand)
    # do this so labels don't get cutoff
    plt.subplots_adjust(bottom=0.25)
    plt.setp(ax.get_xticklabels(),rotation=45)
    plt.title('Brands Most Reviewed')
    plt.xlabel('Brands')
    plt.ylabel('Number of Reviews')
    document.querySelector("#most_tested_graph").innerHTML=""
    display(fig,target="most_tested_graph")
```



PyOhio 2023

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from pyodide.http import open_url
from pyscript import document, display


def show_most_tested_seaborn(event):
    url_content = open_url("https://CSV_URL_HERE")
    df = pd.read_csv(url_content)
    # Remove the unrated entries
    df = df.drop(df[df["Stars"] == "Unrated"].index)
    # Convert to numeric to make it easier to plot
    df["Stars"] = pd.to_numeric(df["Stars"])
    fig,ax = plt.subplots()
    sns.set_theme(style="darkgrid")
    sns.boxplot(data=df,x="Style",y="Stars",palette="pastel",hue="Style").set(title="Reviews by Style")
    document.querySelector("#most_tested_seaborn").innerHTML=""
    display(fig,target="most_tested_seaborn")
```


Reviews by Style
Stars
Style
Cup
Pack
Tray
Bowl
Box
Can
Bar
0
1
2
3
4
5
Style
Cup
Pack
Tray
Bowl
Box
Can
Bar


Brands Most Reviewed
Number of Reviews
400
350
300
250
200
150
100
50
0
Nissin
Nongshim
Maruchan
Mama
Paldo
Myojo
Indomie
Samyang Foods
Ottogi
Lucky Me!

PyOhio 2023

LEARN MORE!

PYSCRIPT EXAMPLES

PyOhio 2023