Will PyScript replace Django?

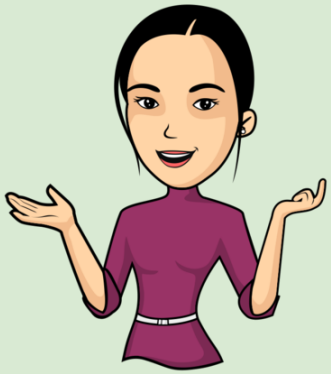
What PyScript is and is not

Cheuk Ting Ho

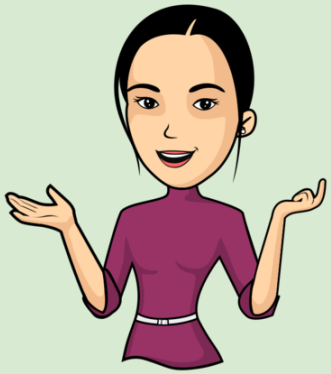
@cheukting_ho Cheukting https://cheuk.dev

Get this slides at
slides.com/cheukting_ho/pyscript-django

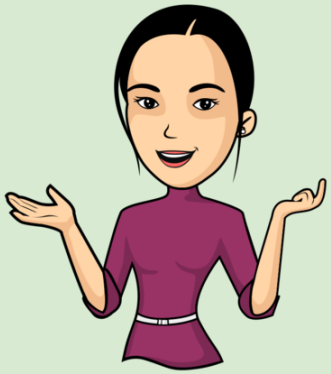
Have you heard of PyScript?



Cheuk Ting Ho

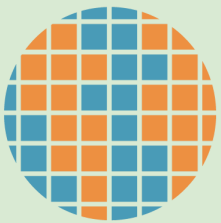


Cheuk Ting Ho

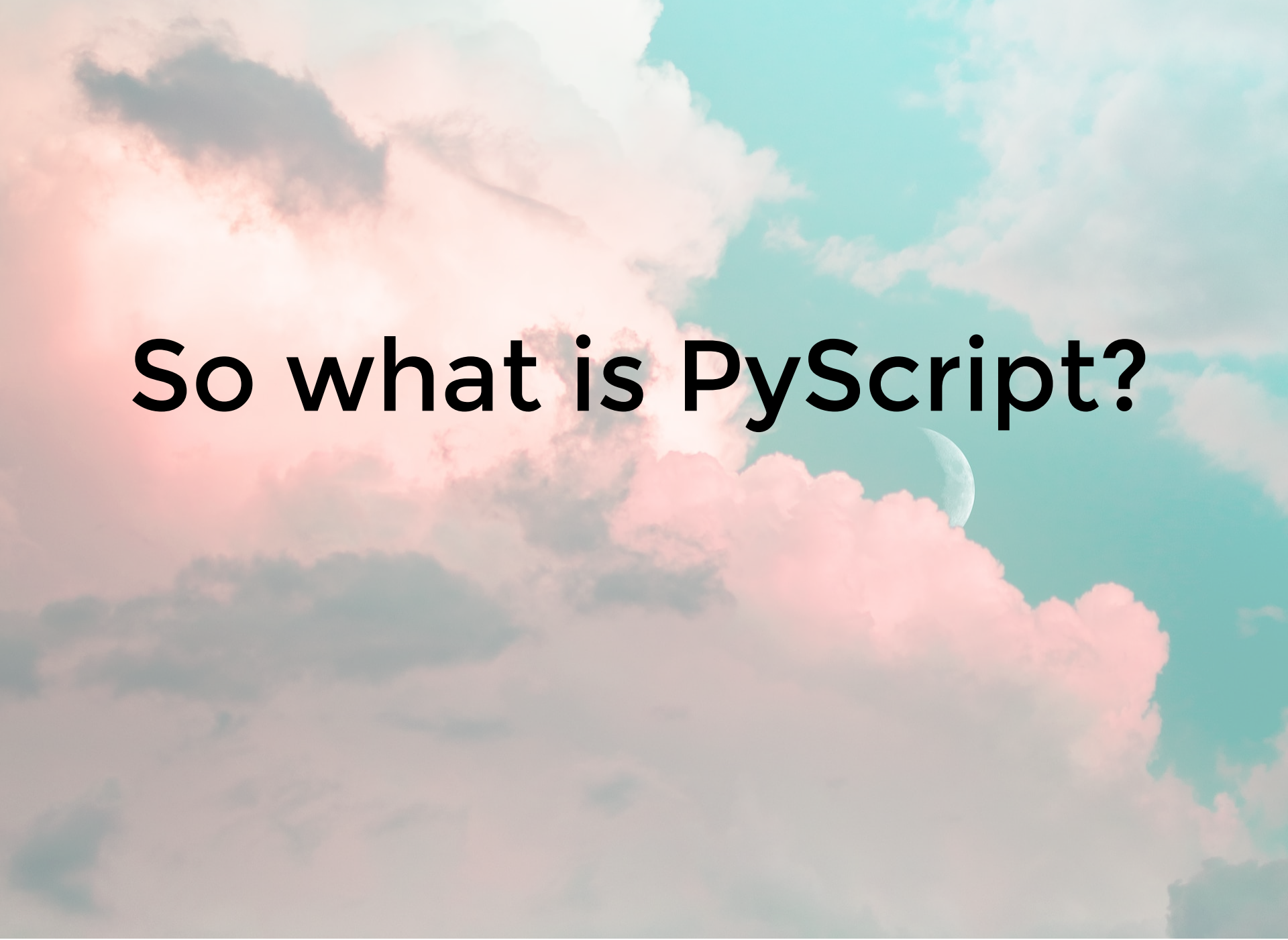


Cheuk Ting Ho

PYTHON
SPRINTS



PyData
Global


So what is PyScript?

So what is PyScript?

So what is PyScript?

- a framework

So what is PyScript?

- a framework
- run Python applications in the browser

So what is PyScript?

- a framework
- run Python applications in the browser
- not to replace JavaScript,
  but rather, you can use with it

So what is PyScript?

- a framework
- run Python applications in the browser
- not to replace JavaScript,
  but rather, you can use with it
- thanks to Pyodide and WASM

So what is PyScript?

- a framework
- run Python applications in the browser
- not to replace JavaScript,
  but rather, you can use with it
- thanks to Pyodide and WASM
- many popular packages of Python avaliable


This is how to use
PyScript

```
<html>
    <head>
      <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
      <script defer src="https://pyscript.net/latest/pyscript.js"></script>
    </head>

  <body>
    <h1>Let's plot random numbers</h1>
    <div id="plot"></div>
    <py-config type="json">
        {
          "packages": ["numpy", "matplotlib"]
        }
    </py-config>
    <py-script output="plot">
      import matplotlib.pyplot as plt
      import numpy as np
      x = np.random.randn(1000)
      y = np.random.randn(1000)
      fig, ax = plt.subplots()
      ax.scatter(x, y)
      fig
    </py-script>
  </body>
</html>
```

```
<html>
    <head>
      <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
      <script defer src="https://pyscript.net/latest/pyscript.js"></script>
    </head>

  <body>
    <h1>Let's plot random numbers</h1>
    <div id="plot"></div>
    <py-config type="json">
        {
          "packages": ["numpy", "matplotlib"]
        }
    </py-config>
    <py-script output="plot">
      import matplotlib.pyplot as plt
      import numpy as np
      x = np.random.randn(1000)
      y = np.random.randn(1000)
      fig, ax = plt.subplots()
      ax.scatter(x, y)
      fig
    </py-script>
  </body>
</html>
```

```
<html>
    <head>
      <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
      <script defer src="https://pyscript.net/latest/pyscript.js"></script>
    </head>

  <body>
    <h1>Let's plot random numbers</h1>
    <div id="plot"></div>
    <py-config type="json">
        {
          "packages": ["numpy", "matplotlib"]
        }
    </py-config>
    <py-script output="plot">
      import matplotlib.pyplot as plt
      import numpy as np
      x = np.random.randn(1000)
      y = np.random.randn(1000)
      fig, ax = plt.subplots()
      ax.scatter(x, y)
      fig
    </py-script>
  </body>
</html>
```

```
<html>
    <head>
      <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
      <script defer src="https://pyscript.net/latest/pyscript.js"></script>
    </head>

  <body>
    <h1>Let's plot random numbers</h1>
    <div id="plot"></div>
    <py-config type="json">
        {
          "packages": ["numpy", "matplotlib"]
        }
    </py-config>
    <py-script output="plot">
      import matplotlib.pyplot as plt
      import numpy as np
      x = np.random.randn(1000)
      y = np.random.randn(1000)
      fig, ax = plt.subplots()
      ax.scatter(x, y)
      fig
    </py-script>
  </body>
</html>
```

```
<html>
    <head>
      <link rel="stylesheet" href="https://pyscript.net/releases/2022.12.1/pyscript.css" />
      <script defer src="https://pyscript.net/releases/2022.12.1/pyscript.js"></script>
    </head>

  <body>
    <h1>Let's plot random numbers</h1>
    <div id="plot"></div>
    <py-config type="json">
        {
          "packages": ["numpy", "matplotlib"]
        }
    </py-config>
    <py-script output="plot">
      import matplotlib.pyplot as plt
      import numpy as np
      x = np.random.randn(1000)
      y = np.random.randn(1000)
      fig, ax = plt.subplots()
      ax.scatter(x, y)
      fig
    </py-script>
  </body>
</html>
```

Or you can download and host it yourself

Just like other JS CDN

The <py-config> tag

The <py-config> tag

toml format
(default)

```
<py-config type="toml">
    packages = ["numpy", "matplotlib"]
    paths = ["./data.py"]
</py-config>
```

The <py-config> tag

toml format
(default)

```
<py-config type="toml">
    packages = ["numpy", "matplotlib"]
    paths = ["./data.py"]
</py-config>
```

```
<py-config type="json">
    {
      "packages": ["numpy", "matplotlib"],
      "paths": ["./data.py"]
    }
</py-config>
```

json format

The <py-config> tag

toml format
(default)

```
<py-config type="toml">
    packages = ["numpy", "matplotlib"]
    paths = ["./data.py"]
</py-config>
```

```
<py-config type="json">
    {
      "packages": ["numpy", "matplotlib"],
      "paths": ["./data.py"]
    }
</py-config>
```

json format

load in config source

```
<py-config type="json" src="./custom.json"></py-config>
```

The <py-config> tag

It is useful for ...

The <py-config> tag

It is useful for ...

- plug in packages (local wheel or hosted)

The <py-config> tag

It is useful for ...

- plug in packages (local wheel or hosted)
- load in local modules (fetch)

The <py-config> tag

It is useful for ...

- plug in packages (local wheel or hosted)
- load in local modules (fetch)
- change runtime settings (e.g. setting runtime source)

The <py-config> tag

It is useful for ...

- plug in packages (local wheel or hosted)
- load in local modules (fetch)
- change runtime settings (e.g. setting runtime source)
- adding meta data (e.g. decription, author_name, license)

The <py-repl> tag

Interactive Python interface (like jupyter notebook)

```
<html>
  <head>
    <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
    <script defer src="https://pyscript.net/latest/pyscript.js"></script>
  </head>
  <py-repl></py-repl>
</html>
```

The <py-repl> tag

Interactive Python interface (like jupyter notebook)

```
<html>
  <head>
    <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
    <script defer src="https://pyscript.net/latest/pyscript.js"></script>
  </head>
  <py-repl></py-repl>
</html>
```

Why is it useful?

a.k.a why do it on frontend

Sometime we need to use frontend because...

Sometime we need to use frontend because...

- Computation is heavy

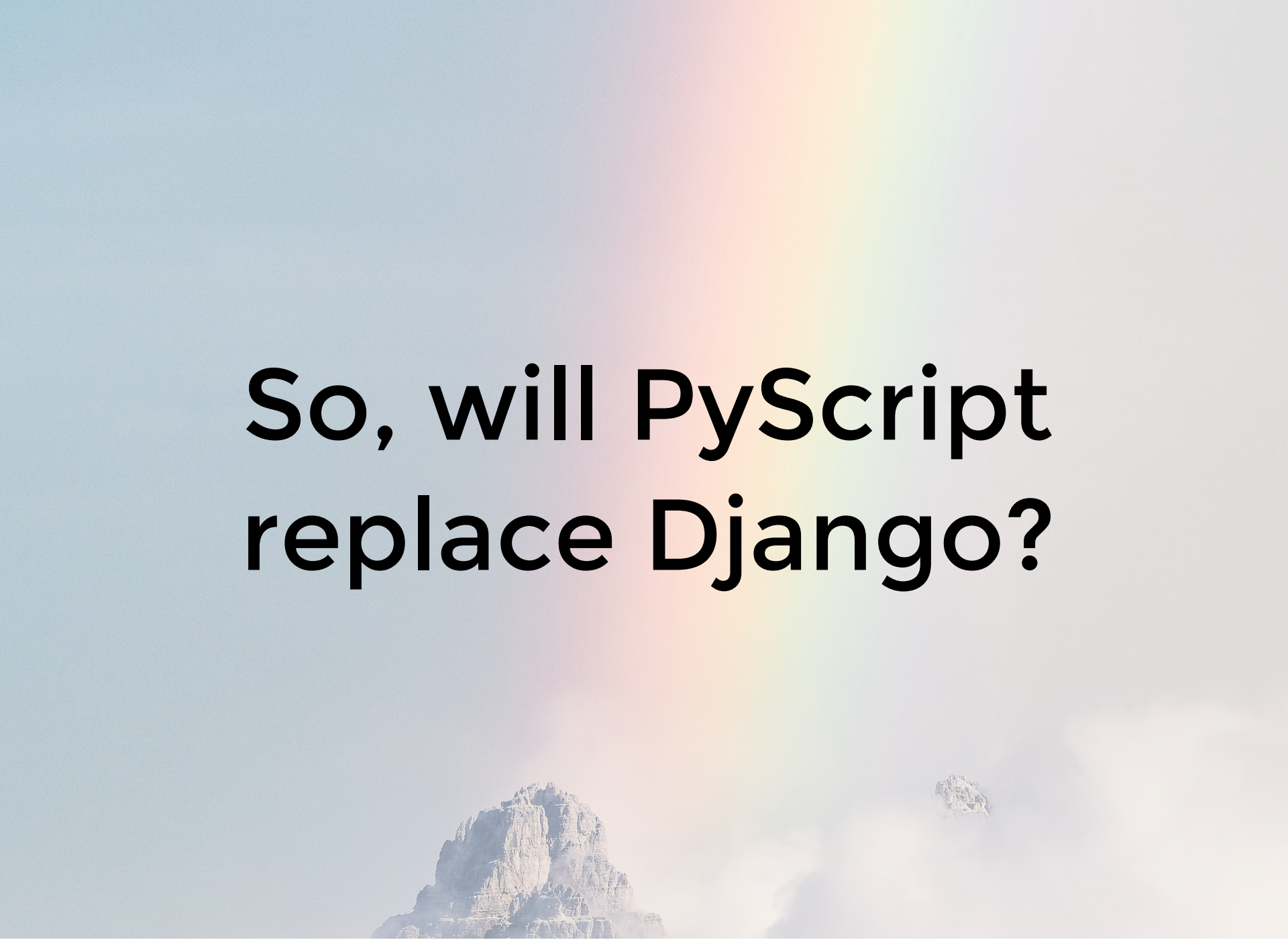
Sometime we need to use frontend because...

- Computation is heavy
- Don't want to provide a playground (crypto mining)

Sometime we need to use frontend because...

- Computation is heavy
- Don't want to provide a playground (crypto mining)
- Data too sensitive to leave user's own machine

So, will PyScript replace Django?




No,
but we can make some amazing stuff if we use them together 🙌

Here is an example

Using PyScript with Django

https://github.com/Cheukting/django-knn-recommender

Other examples

Running Django with Pyscript - Frontend as a Backend?!

https://absurd-django.vercel.app/

by Patrick Arminio

More Pyscript examples...

- Interactive plot
  https://cheuk.dev/assets/html/interactive_network_graph.html
- Using Follim for map plotting
  https://pyscript.net/examples/folium.html
- Interactive plot with d3 (We can now `import d3`)
  https://cheuk.dev/assets/html/ice-cream-d3.html

PyScript Tutorial [WIP]:

https://github.com/anaconda/pyscript-tutorial

Q & A

Q & A

- Can you pull in a python script?
  - Yes now you can with [[fetch]]

Q & A

- Can you pull in a python script?
  - Yes now you can with [[fetch]]

- What is the Python version we are using?
  - Choose your version with the `src` var at [[runtimes]]

Q & A

- Can you pull in a python script?
  - Yes now you can with [[fetch]]

- What is the Python version we are using?
  - Choose your version with the `src` var at [[runtimes]]

- Why can't you use `<script tag="python">`
  - Because `<script>` is not a custom tag and it is defined by the browser which none of them is supporting Python yet

- Why don't you just use Pyodide?
  - It's like we use Keras to simplify using TensorFlow, PyScript make it easier to using Python on the browser

Q & A

Q & A

- Can you pin a package version at <py-config>?
  - Right now you cannot, but you can host the wheel yourself (see example here)

- What is the difference between PyScirpt and Brython?
  - PyScript is interpreting Python to WASM instead of JS and has many popular packages available
  - You can also pick which Python WASM backend to use

- Any plan to support BeeWare?
  - I would love to personally, we will see