How to be Pythonic?
Design a Query Language in Python

Cheuk Ting Ho
https://cheuk.dev  🐍@cheuktting_ho  🐱Cheukting

Grab the slides: bit.ly/be-pythonic
Cheuk Ting Ho

Pick & Mix

Eye

TerminusDB

Python Sprints

 Indie Web Camp
What does Pythonic mean? (Is it a thing?)
What does Pythonic mean? (Is it a thing?)

Is Pythonic a thing?

- Yes: 88.6%
- No: 11.4%
“Pythonic means code that doesn't just get the syntax right but that follows the conventions of the Python community and uses the language in the way it is intended to be used."
“Pythonic means code that doesn't just get the syntax right but that follows the conventions of the Python community and uses the language in the way it is intended to be used.

- Stackoverflow
"Pythonic means code that doesn't just get the syntax right but that follows the conventions of the Python community and uses the language in the way it is intended to be used."

- Stackoverflow
“Pythonic means code that doesn't just get the syntax right but that follows the conventions of the Python community and uses the language in the way it is intended to be used.”

- Stackoverflow

Why can't I just do it in a for-loop?
for i in (i; i < items.length ; i++)
{
    n = items[i];
    ... now do something
}

for i in (i; i < items.length ; i++)
{
  n = items[i];
  ... now do something
}

for i in items:
  i.perform_action()
for i in (i; i < items.length ; i++) {
  n = items[i];
  ... now do something
}

for i in items:
  i.perform_action()

(i.some_attribute for i in items)
```python
for i in items:
  i.perform_action()
```

Pythonic!
Design a Query Language in Python
Design a Query Language in Python

Who like SQL?

I love it: 68.4%
No way: 31.6%
It all stated...

Developer Advocate of TerminusDB
<table>
<thead>
<tr>
<th>Person_ID</th>
<th>Name</th>
<th>DOB</th>
<th>mother</th>
<th>father</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>John</td>
<td>1/10/79</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Mary</td>
<td>4/2/56</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>John snr</td>
<td>28/11/52</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>Patricia</td>
<td>17/4/22</td>
<td>null</td>
<td>null</td>
</tr>
<tr>
<td>5</td>
<td>Michael</td>
<td>1/9/09</td>
<td>null</td>
<td>null</td>
</tr>
<tr>
<td>6</td>
<td>Sally</td>
<td>17/4/23</td>
<td>null</td>
<td>null</td>
</tr>
<tr>
<td>7</td>
<td>Robert</td>
<td>3/10/13</td>
<td>null</td>
<td>null</td>
</tr>
</tbody>
</table>
1. SELECT Name from TABLE where Person_ID = (SELECT mother from TABLE where Name="John")
2. SELECT Name from TABLE where Person_ID = (SELECT mother from TABLE WHERE Person_ID = (SELECT mother from TABLE where Name="John"))
WOQL.and(
   WOQL.triple("v:Person", "mother", "v:MotherID"),
   WOQL.triple("v:MotherID", "name", "v:MotherName"),
   WOQL.triple("v:MotherID", "mother", "v:GrandmotherID"),
   WOQL.triple("v:GrandmotherID", "name", "v:GrandmotherName"),
)

1
2
3
4
5
6
WOQLpy
a Query Language Client for Pythonistas and Data Scientists
WOQLpy

a Query Language Client for Pythonistas
and Data Scientists
What is WOQLpy?
It comes with the Python Client, which you can pip install:

```
1 pip install terminusdb-client
```
It comes with the Python Client, which you can pip install:

```
pip install terminusdb-client
```

**Newly added** Output to DataFrames

```
pip install terminusdb-client[dataframe]
```

Change the result returned form your query into pandas DataFrame

```
woql.query_to_df(result)
```
It lets you to "talk" to TerminusDB like this:

```python
import terminusdb_client as woql
from terminusdb_client import WOQLQuery

db_id = "pybike"
client = woql.WOQLClient(server_url = "http://localhost:6363")
client.connect(key="root", account="admin", user="admin")
client.create_database(db_id, accountid="admin", label = "Bike Graph",
                      description = "Create a graph with bike data")

station_dt = WOQLQuery().doctype("Station",
                                label="Bike Station",
                                description="A station where bikes are deposited")
bicycle_dt = WOQLQuery().doctype("Bicycle", label="Bicycle")
journey_dt = (WOQLQuery().doctype("Journey", label="Journey").
              property("start_station", "Station", label="Start Station").
              property("end_station", "Station", label="End Station").
              property("duration", "integer", label="Journey Duration").
              property("start_time", "dateTime", label="Time Started").
              property("end_time", "dateTime", label="Time Ended").
              property("journey_bicycle", "Bicycle", label="Bicycle Used")
            )

schema = station_dt + bicycle_dt + journey_dt
schema.execute(client)
```
Instead of this:

```json
{
  "when": [
    {
      "true": []
    },
    {
      "and": [
        {
          "add_quad": [
            "scm:Station",
            "rdf:type",
            "owl:Class",
            "db:schema"
          ]
        },
        {
          "add_quad": [
            "scm:Station",
            "rdfs:subClassOf",
            "tcs:Document",
            "db:schema"
          ]
        }
      ]
    }
  ]
}
```
You can do both

```
(WOQLQuery().doctype("Station")
  .label=("Bike Station")
  .description("A station where bikes are deposited")
)
```

or

```
WOQLQuery().doctype("Station",
  label="Bike Station",
  description="A station where bikes are deposited")
```
Which one do you prefer?

Chaining:
WOQLQuery().doctype("journey").label("Journey")

or

Multi-parameters:
WOQLQuery().doctype("journey, label="Journey")
Which one do you prefer?

Chaining:
WOQLQuery().doctype("journey").label("Journey")

or

Multi-parameters:
WOQLQuery().doctype("journey, label="Journey")

Which one do you prefer?

<table>
<thead>
<tr>
<th>Multi-parameters</th>
<th>76%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chaining</td>
<td>24%</td>
</tr>
</tbody>
</table>
Design challenges

JavaScript: WOQL.and()

Python: WOQLQuery().and() ?
Design challenges

JavaScript: WOQL.and()

Python: WOQLQuery().and()

"and" is a key word, you dummy!
Design challenges

JavaScript: WOQL.and()

Python: WOQLQuery().and()

"and" is a key word, you dummy!

OK, woql_and then....😢
WOQLQuery().woql_and()

* actually you can use the + operator
thanks for the overload ability in Python

also happened to: or, not, as, from...
Look into the future
<table>
<thead>
<tr>
<th>End</th>
<th>Start</th>
<th>Start Label</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>S Glebe &amp; Potomac Ave <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Crystal City Metro / 18th &amp; Bell St</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>S Glebe &amp; Potomac Ave <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Crystal City Metro / 18th &amp; Bell St</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>New Hampshire Ave &amp; T St NW <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>14th &amp; V St NW</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>New Hampshire Ave &amp; T St NW <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>14th &amp; V St NW</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Georgia Ave and Fairmont St NW <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>14th &amp; V St NW</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>11th &amp; O St NW <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>New Jersey Ave &amp; N St NW/Dunbar HS</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>11th &amp; O St NW <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>New Jersey Ave &amp; N St NW/Dunbar HS</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>D St &amp; Maryland Ave NE <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>2nd St &amp; Massachusetts Ave NE</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Maine Ave &amp; 7th St SW <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Potomac Ave &amp; Half St SW</td>
</tr>
<tr>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>King Farm Blvd &amp; Piccard Dr <a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Shady Grove Metro West</td>
</tr>
</tbody>
</table>

1862 rows x 4 columns
Can we have a nice graph visualization?

<table>
<thead>
<tr>
<th>End</th>
<th>End_Label</th>
<th>Start</th>
<th>Start_Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>S Glebe &amp; Potomac Ave</td>
<td>Crystal City Metro / 18th &amp; Bell St</td>
</tr>
<tr>
<td>1</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>S Glebe &amp; Potomac Ave</td>
<td>Crystal City Metro / 18th &amp; Bell St</td>
</tr>
<tr>
<td>2</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>New Hampshire Ave &amp; T St NW</td>
<td>14th &amp; V St NW</td>
</tr>
<tr>
<td>3</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>New Hampshire Ave &amp; T St NW</td>
<td>14th &amp; V St NW</td>
</tr>
<tr>
<td>4</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Georgia Ave and Fairmont St NW</td>
<td>14th &amp; V St NW</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>1857</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>11th &amp; C St NW</td>
<td>New Jersey Ave &amp; N St NW/Dunbar HS</td>
</tr>
<tr>
<td>1858</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>11th &amp; C St NW</td>
<td>New Jersey Ave &amp; N St NW/Dunbar HS</td>
</tr>
<tr>
<td>1859</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>D St &amp; Maryland Ave NE</td>
<td>2nd St &amp; Massachusetts Ave NE</td>
</tr>
<tr>
<td>1860</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>Maine Ave &amp; 7th St SW</td>
<td>Potomac Ave &amp; Half St SW</td>
</tr>
<tr>
<td>1861</td>
<td><a href="http://localhost:5363/pybike/document/Station3">http://localhost:5363/pybike/document/Station3</a>...</td>
<td>King Farm Blvd &amp; Piccard Dr</td>
<td>Shady Grove Metro West</td>
</tr>
</tbody>
</table>

1862 rows x 4 columns
Load data from DataFrame 🐼
Load data from DataFrame

CLI client (with click?)
Load data from DataFrame 🐼

CLI client (with click?)

Many more fail-proof checks
 e.g. check user inputs,
 check database version etc...
World of WoqlCraft:
Every Friday 5pm UK time / 6pm CET

To get the newest update👍:
Follow us on Twitter: @TerminusDB
Website: https://terminusdb.com/

Join the community at Discord:
https://discord.gg/Gvdqw97

We want to hear from you 😊