



PyCon Shanghai 2019

科技助力美好生活
Technology for a better life

Follow me on Twitter [@dglover](#)

Dave Glover,
Microsoft
Cloud Advocate



我喜欢Python的理由



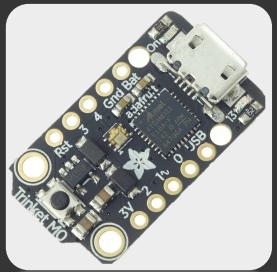
Apollo Guidance Software

<https://www.vox.com/2015/5/30/8689481/margaret-hamilton-apollo-software>



CubeSat
Raspberry Pi & Python

Python无处不在



嵌入式



教育

Web开发

自动化



数据分析

机器学习



Python, 开源, 免
费开始使用 Azure

Python, Open Source, get started on Azure for free

视障人士购物

Shopping for vision impaired people

方案概述 Solution Overview

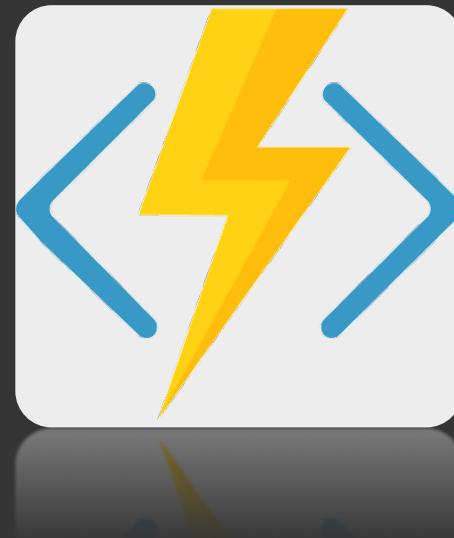
智能边缘

Intelligent Edge



云集成

Cloud Integration

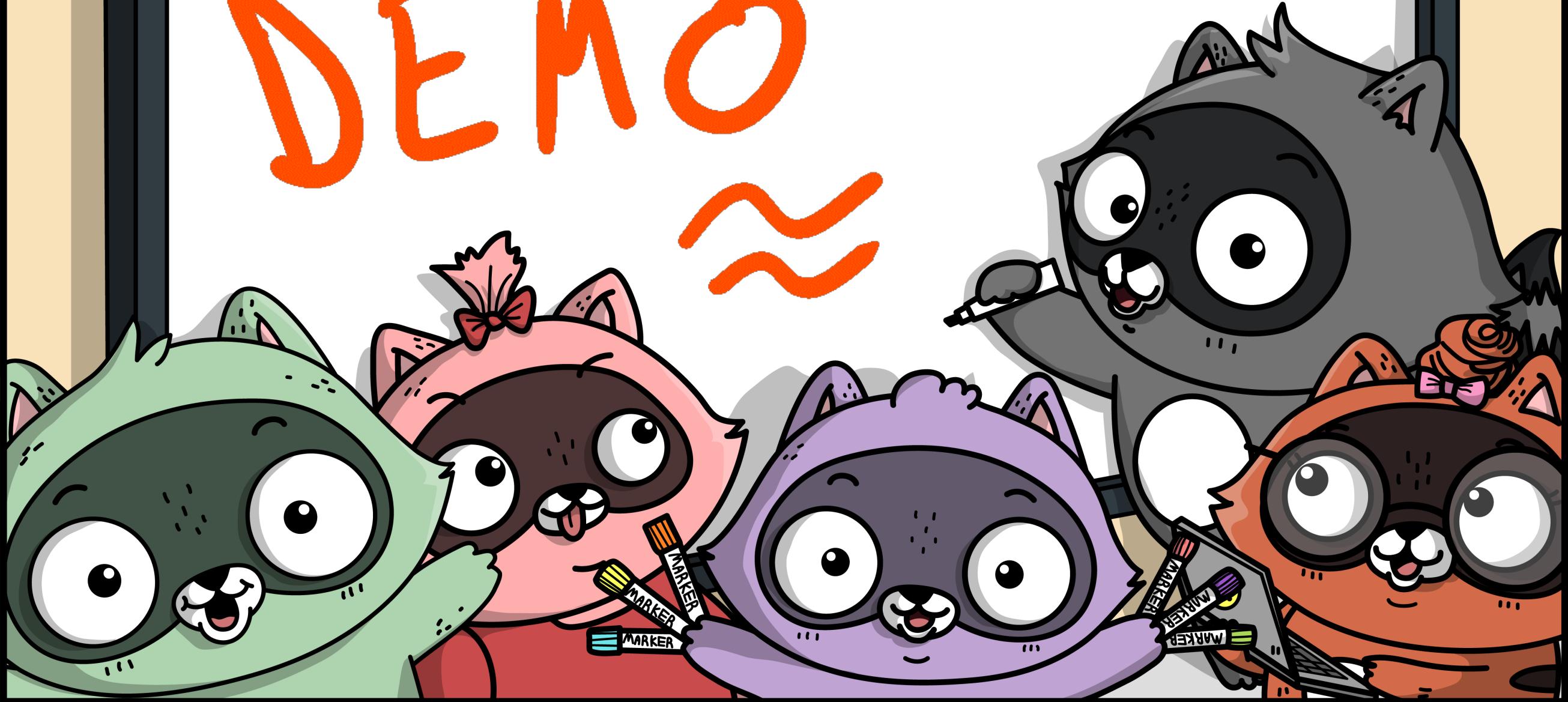


数据分析

Data Analysis



DEMO





Azure 自定义视觉



Azure Custom Vision

The screenshot shows the Azure Custom Vision results interface. On the left, there is a thumbnail image of two oranges. To the right, a larger image of a whole orange and a slice of orange is displayed. Below these images, a table lists the results:

Tag	Probability
Orange	99.9%
Red Apple	3.1%
Green Apple	0.1%
Banana	0.0%

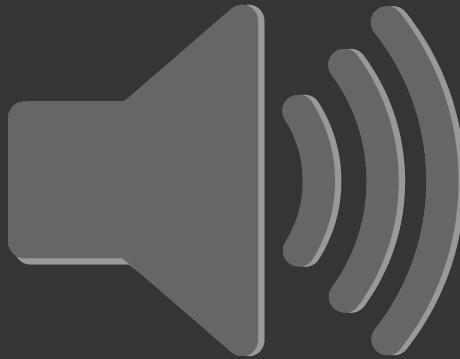
Azure语音服务



Azure Speech Service

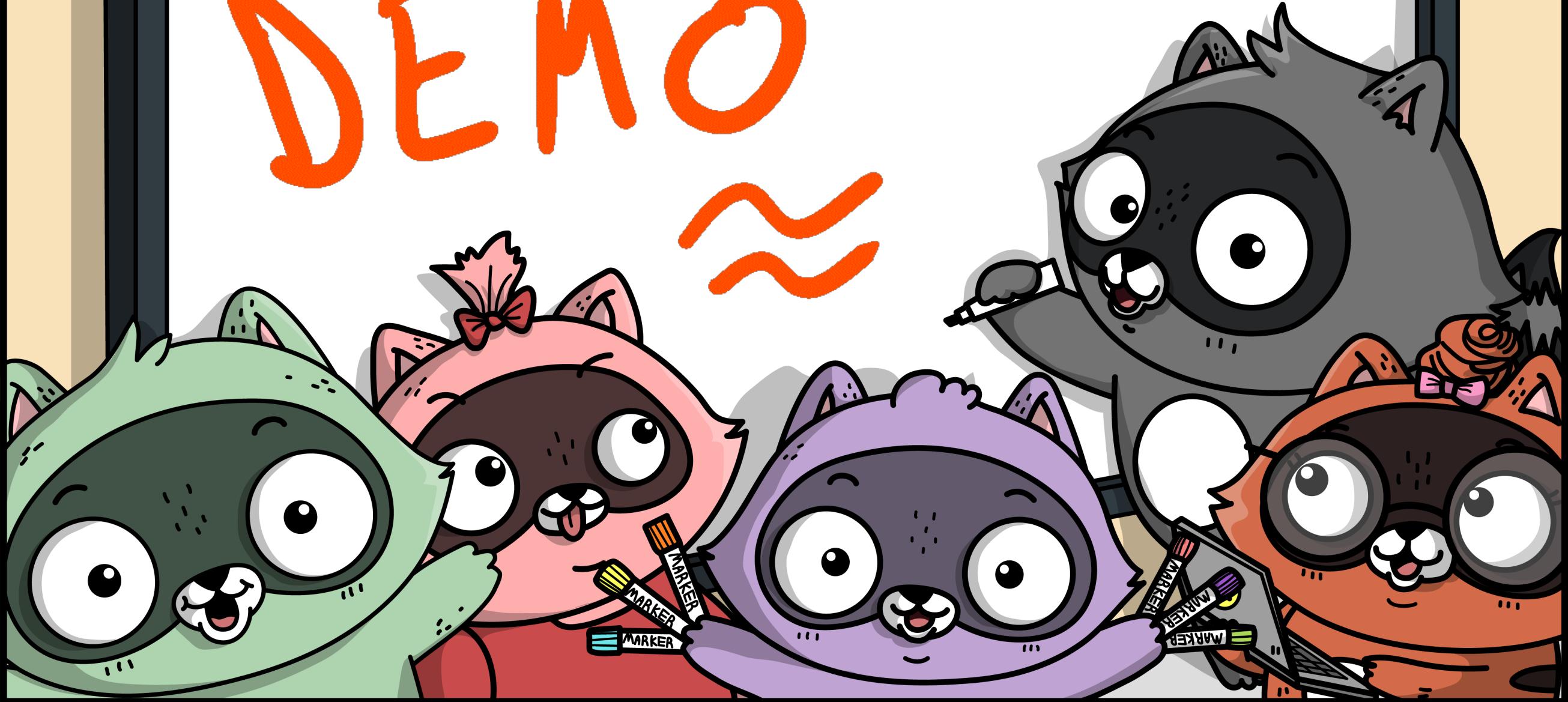


语音翻译



文字转语音

DEMO



Filter

Iteration

Workspace

Tags

 Tagged Untagged

Showing: all tagged images

Search For Tags:

- Banana 55 ...
- Green Apple 180 ...
- Hand 10 ...
- Orange 119 ...
- Red Apple 180 ...

Add images

Delete

Tag images

Select all

< 1 2 3 4 5 >



Get started



EXPLORER

predict.py

≡ README.txt ×



OPEN EDITORS

predict.py app

X ≡ README.txt



PYCON-CUSTOM-VISION [...]

> .vscode

✓ app

> __pycache__

app.py 4

≡ labels.txt

≡ model.pb

predict.py

≡ requirements.txt M

> azureml

docke-compose.yml

Dockerfile

≡ README.txt



≡ README.txt

1 How to build:

2 =====

3

4 docker build -t <your image name> .

5

6 How to run locally:

7 =====

8

docker run -p 127.0.0.1:80:80 -d <your image name>

9

10 Then use your favorite tool to connect to the end points.

11

12 POST <http://127.0.0.1/image> with multipart/form-data using the imageData

13 e.g

14 curl -X POST <http://127.0.0.1/image> -F imageData=@some_file_name.jpg

15

16 POST <http://127.0.0.1/image> with application/octet-stream

17 e.g.

18 curl -X POST <http://127.0.0.1/image> -H "Content-Type: application/oc

19



方案概述 Solution Overview

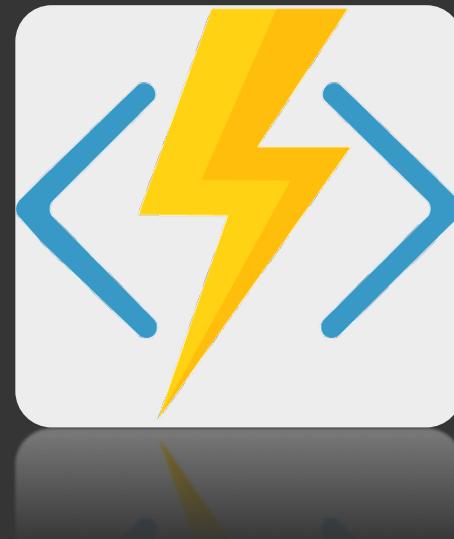
智能边缘

Intelligent Edge



云集成

Cloud Integration



数据分析

Data Analysis



云集成与Azure功能

Cloud Integration with Azure Functions

智能边缘

Intelligent Edge



Azure IoT



Python Azure
功能



Azure 存储

Azure Storage



tagName	Count	probability
Green Apple	37	0.96
Red Apple	56	0.99
Orange	68	0.92
Banana	87	0.97
Apple	81	0.60
Orange	60	0.25

Python Azure Functions



没有基础设施问题
No infrastructure headaches



按需扩展
Scales on demand



运行成本低
Very cheap to run

Python Azure Functions

The screenshot shows the Azure portal interface for managing a Python Azure Function. The left sidebar contains various icons for different services like Storage, Cosmos DB, and App Services. The main area displays the function's configuration and code.

Function Apps: enviromon-python - EnvironmentEventTrigger

Code Editor: __init__.py

```
1 import logging
2 import azure.functions as func
3 from azure.cosmosdb.table.tableservice import TableService
4 import requests
5 import json
6 import os
7 import random
8 import time
9 from ..SharedCode import calibrate
10
11 # https://azure.microsoft.com/en-au/blog/managing-concurrency-in-microsoft-azure-storage-2/
12 # https://docs.microsoft.com/en-us/python/api/azure-cosmosdb-table/azure.cosmosdb.table.tableservice.tab
13
14 deviceStateTable = "DeviceState"
15 calibrationTable = "Calibration"
16
17 storageConnectionString = os.environ['StorageConnectionString']
18 partitionKey = os.environ.get('PartitionKey', 'Sydney')
19 signalrUrl = os.environ['SignalrUrl']
20
21 table_service = TableService(connection_string=storageConnectionString)
22 if not table_service.exists(deviceStateTable):
23     table_service.create_table(deviceStateTable)
24
25 calibrator = calibrate.Calibrate(table_service, calibrationTable, partitionKey)
26
27 # Optimistic Concurrency Tuning Parameters
```

Logs:

方案概述 Solution Overview

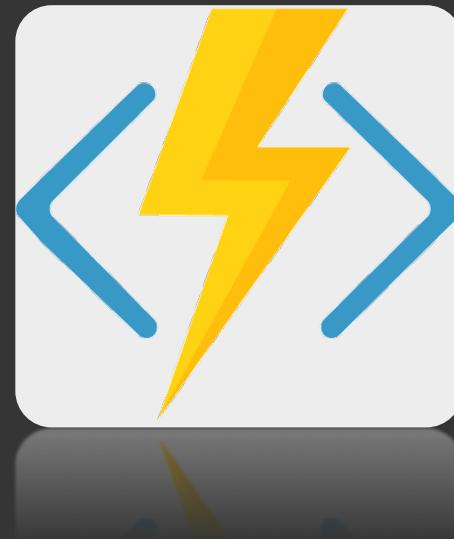
智能边缘

Intelligent Edge



云集成

Cloud Integration

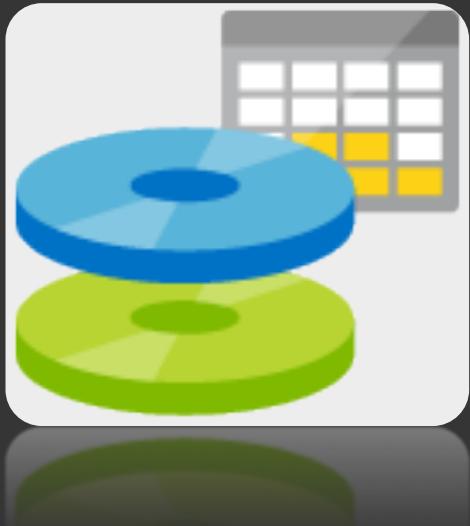


数据分析

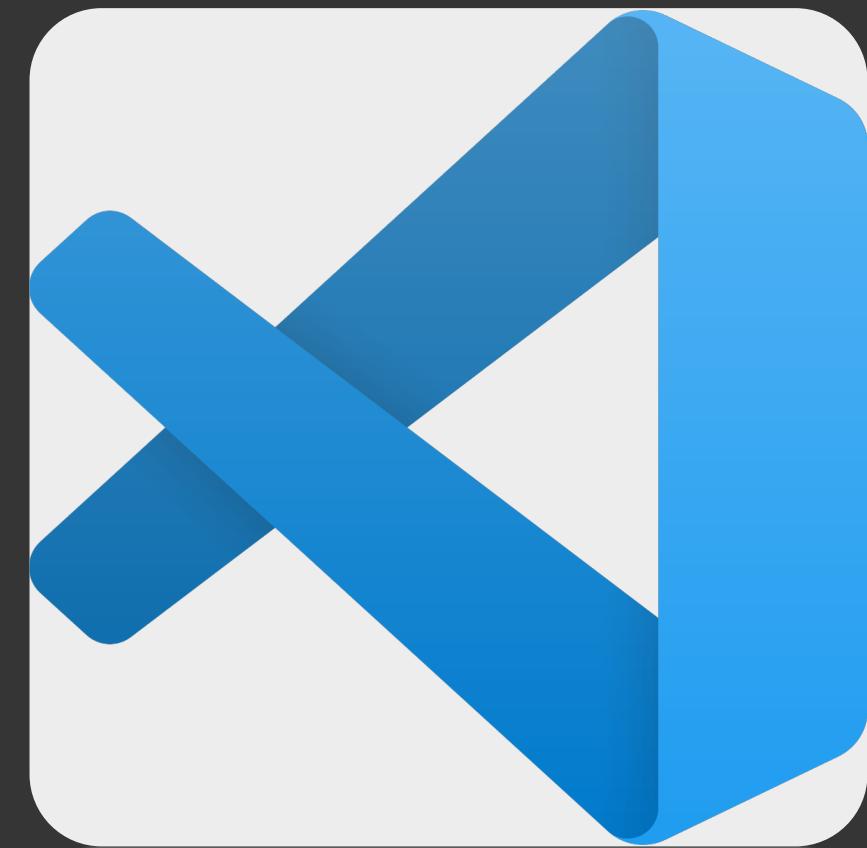
Data Analysis



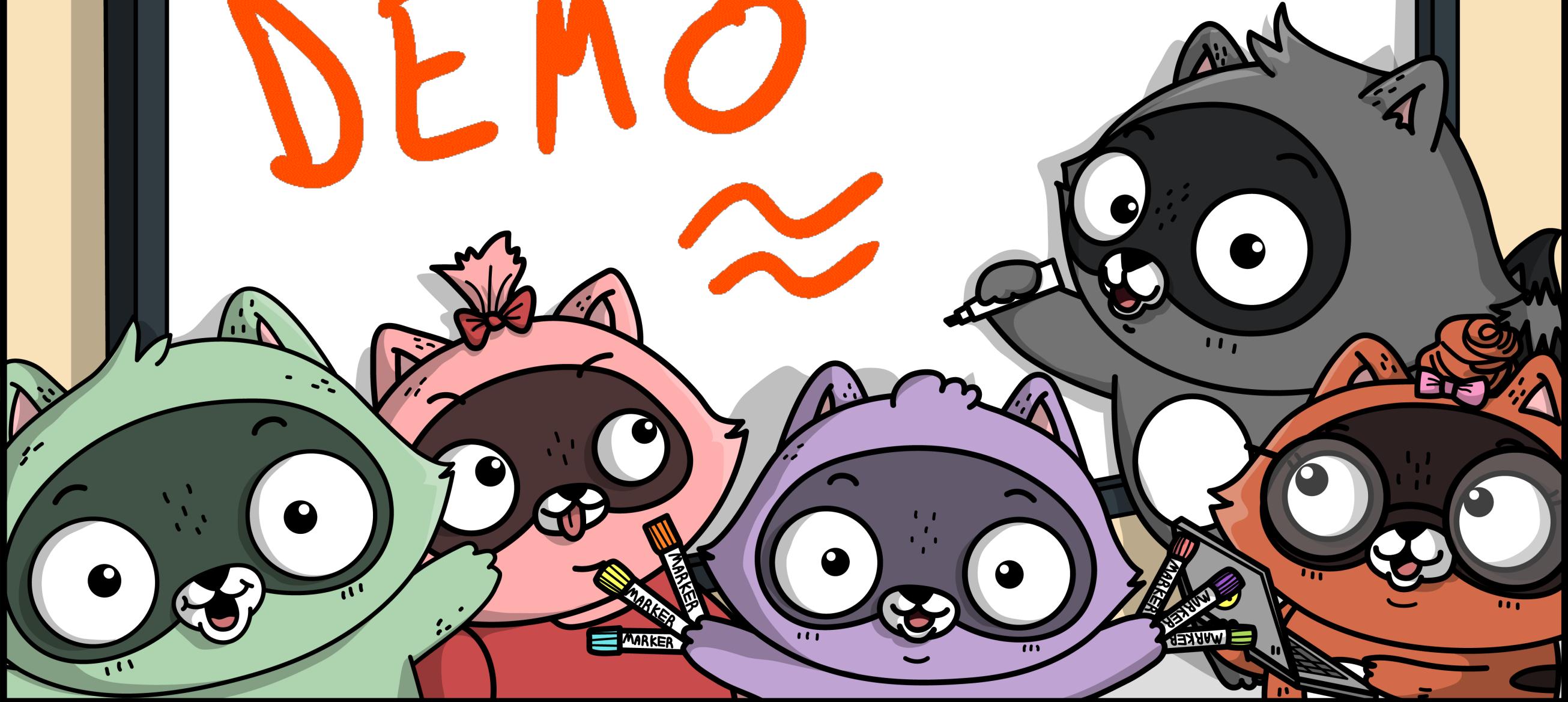
使用Jupyter Notebook进行数据分析



tagName	Count	probability
Green Apple	37	0.96
Red Apple	56	0.99
Orange	68	0.92
Banana	87	0.97



DEMO



jupyter image-scanning-data

Last Checkpoint: an hour ago (unsaved changes)



Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted

Python 3



```
Requirement already satisfied: cffi!=1.11.3,>=1.8 in c:\users\dglover\appdata\local\programs\python\python36\lib\site-packages (from cryptocrpto>azure-cosmosdb-table) (1.12.3)
```

```
In [2]: table_name = 'IoTEdgeImageClassifier'
storageConnectionString =
    "DefaultEndpointsProt
```

```
In [ ]: from azure.cosmosdb.table.tableservice import TableService
from matplotlib import pyplot as plt
```

```
In [ ]: table_service = TableService(connection_string=storageConnectionString)

# Query the table
queried_entities = table_service.query_entities(table_name, filter="PartitionKey eq 'Register1'")
data = [i for i in queried_entities]
print(data)
```

```
In [ ]: x = []
y = []

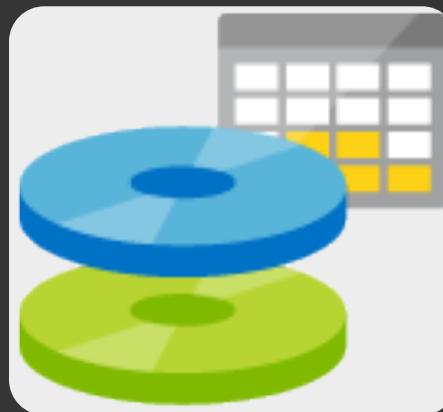
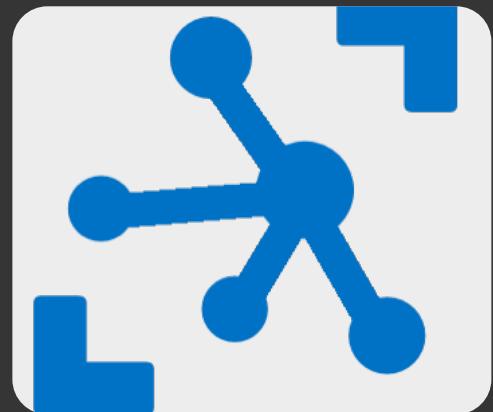
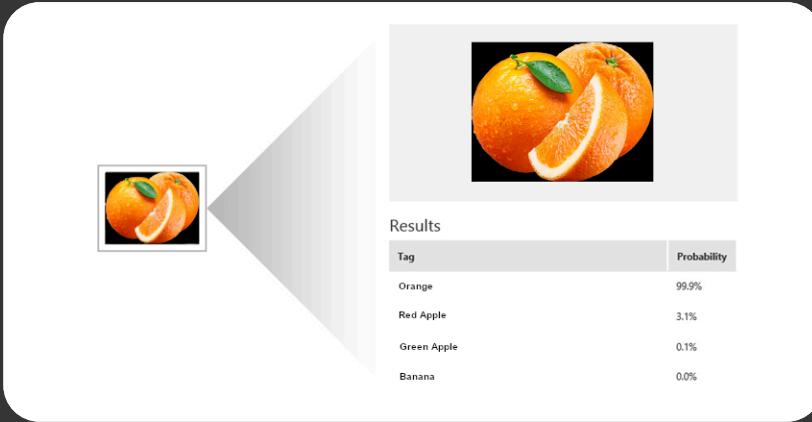
for item in data:
    x.append(item.get('tag'))
    y.append(item.get('Count'))

plt.bar(x, y)
plt.show()
```

```
In [ ]:
```

```
In [ ]:
```

Python端到端



立刻行动

Call To Action

- Learn something new at PyCon
- Meet new people at PyCon
- Find a fun Project
 - <https://www.hackster.io/>
- Blog and share what you learn
- Read the docs
 - <https://docs.python.org/3.7/> (Select Simplified Chinese)

资源

<http://aka.ms/PyConShanghai>

謝謝