Microsoft

# PyCon Shanghai 2019

## 科技助力美好生活
## Technology for a better life

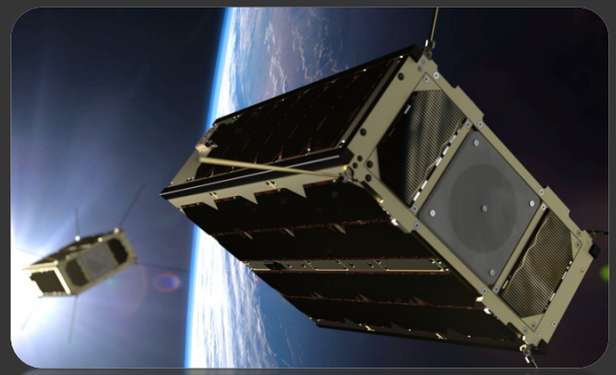

Follow me on Twitter **@dglover**

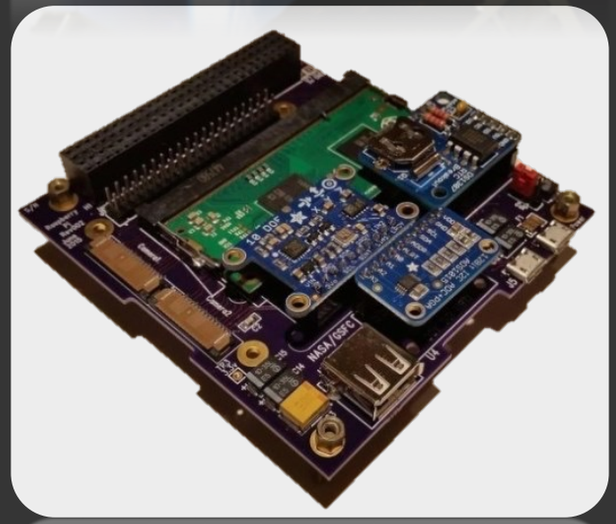
Dave Glover,
Microsoft
Cloud Advocate

# 我喜欢Python的理由

Apollo Guidance Software

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



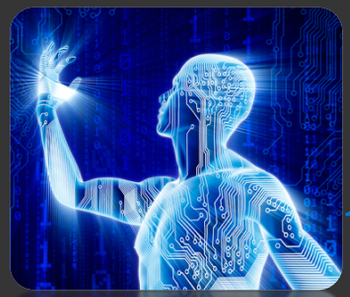


CubeSat

Raspberry Pi & Python

# Python无处不在

数据分析

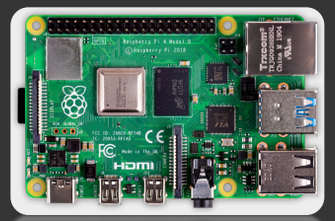


机器学习

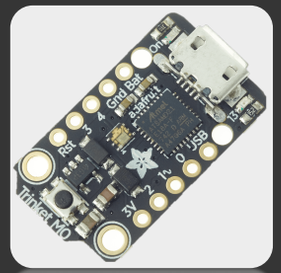
自动化

Web开发

教育



嵌入式



# 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**

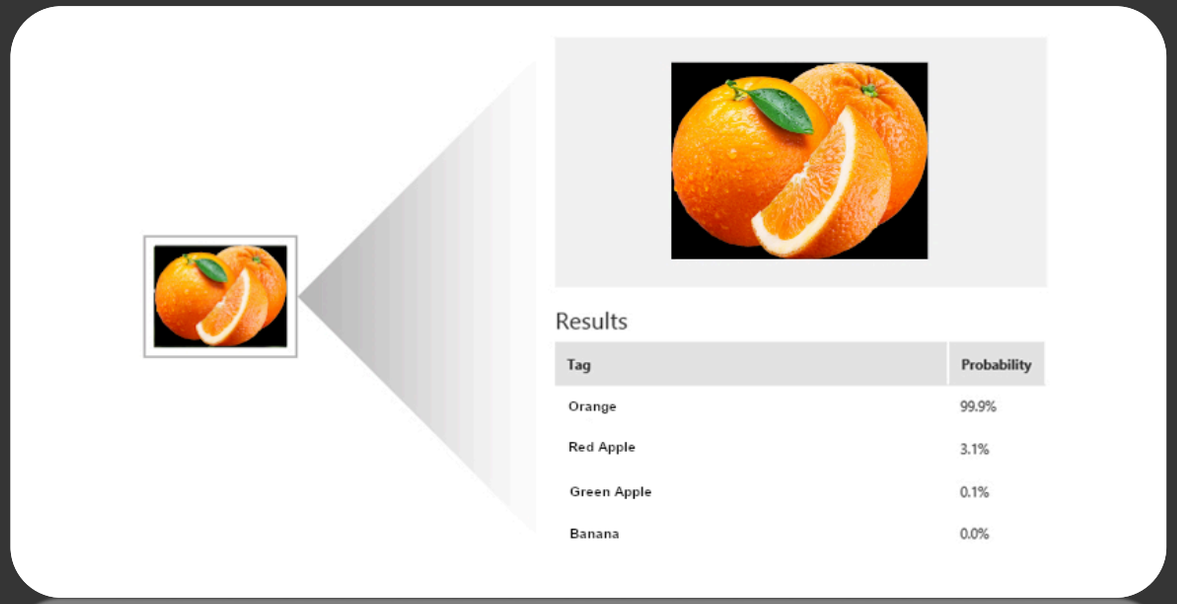
数据分析
**Data Analysis**


DEMO
MARKER
MARKER
MARKER
MARKER
MARKER
MARKER


Microsoft

# Azure自定义视觉

# Azure Custom Vision



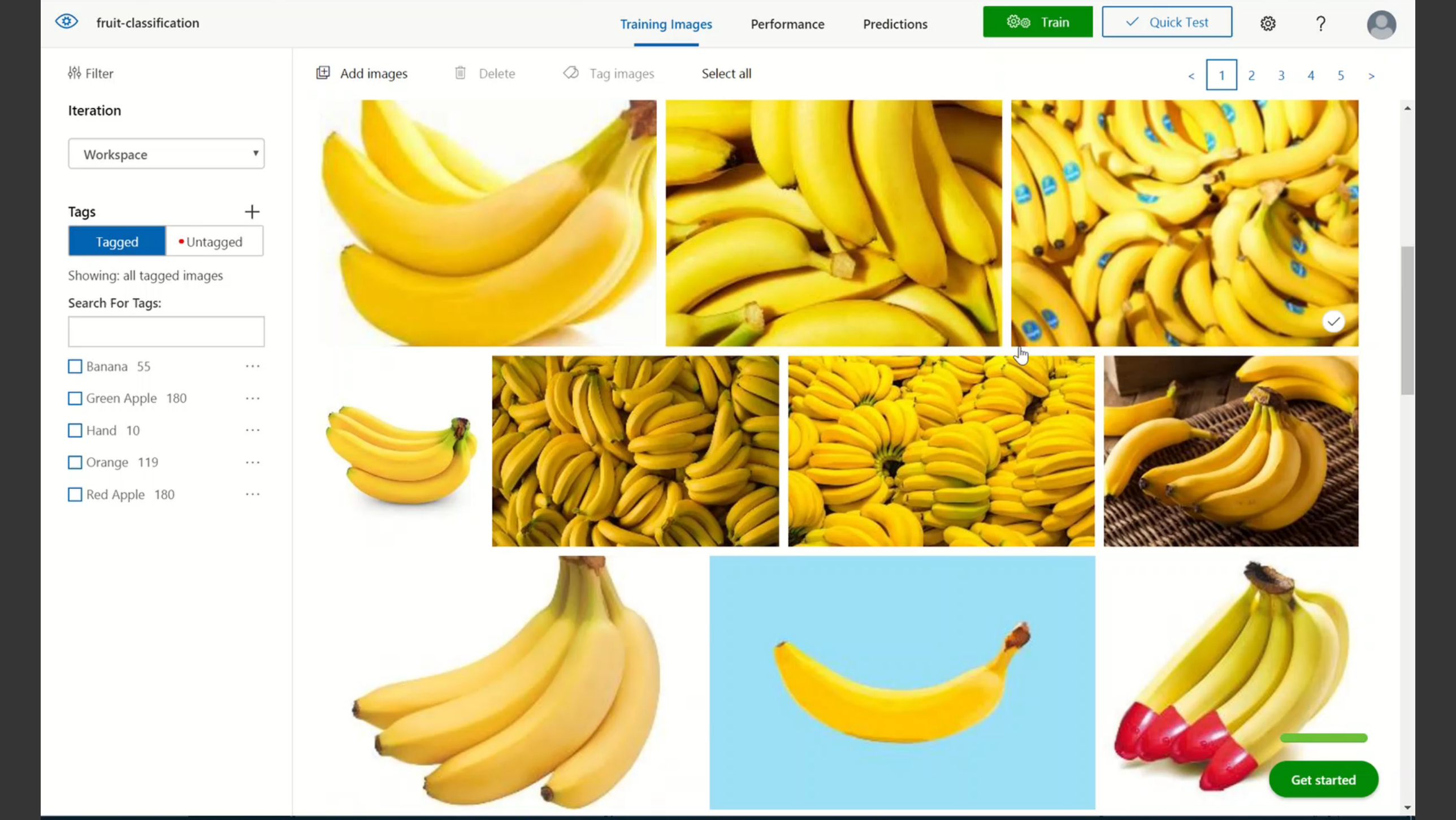
# Azure语音服务

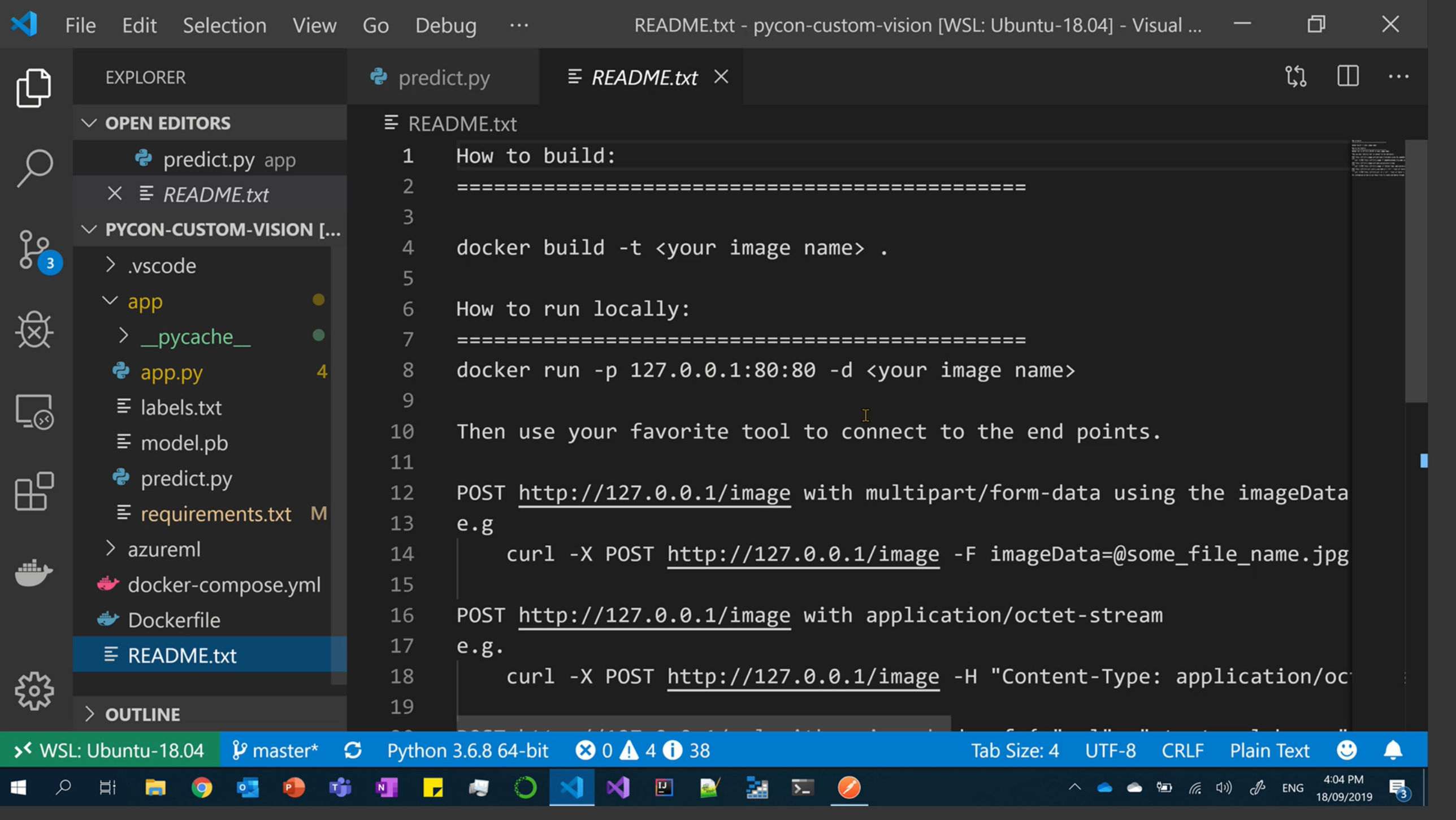
# Azure Speech Service

语音翻译

文字转语音


DEMO
MARKER
MARKER
MARKER
MARKER
MARKER
MARKER


fruit-classification
Training Images
Performance
Predictions
Train
Quick Test
Filter
Add images
Delete
Tag images
Select all
1
2
3
4
5
Iteration
Workspace
Tags
Tagged
Untagged
Showing: all tagged images
Search For Tags:
Banana 55
Green Apple 180
Hand 10
Orange 119
Red Apple 180
Get started


File Edit Selection View Go Debug ...
README.txt - pycon-custom-vision [WSL: Ubuntu-18.04] - Visual ...
EXPLORER
predict.py
README.txt
OPEN EDITORS
predict.py app
README.txt
PYCON-CUSTOM-VISION [...
.vscode
app
__pycache__
app.py 4
labels.txt
model.pb
predict.py
requirements.txt M
azureml
docker-compose.yml
Dockerfile
README.txt
OUTLINE
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
WSL: Ubuntu-18.04
master*
Python 3.6.8 64-bit
0 4 38
Tab Size: 4
UTF-8
CRLF
Plain Text
ENG
4:04 PM
18/09/2019

# 方案概述 Solution Overview

智能边缘
**Intelligent Edge**

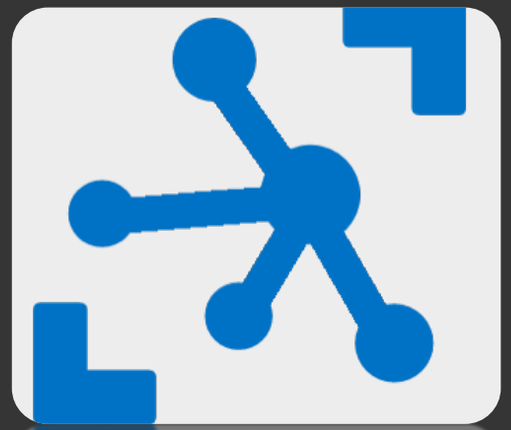

云集成
**Cloud Integration**

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 |

# Python Azure Functions

没有基础设施问题

No infrastructure headaches

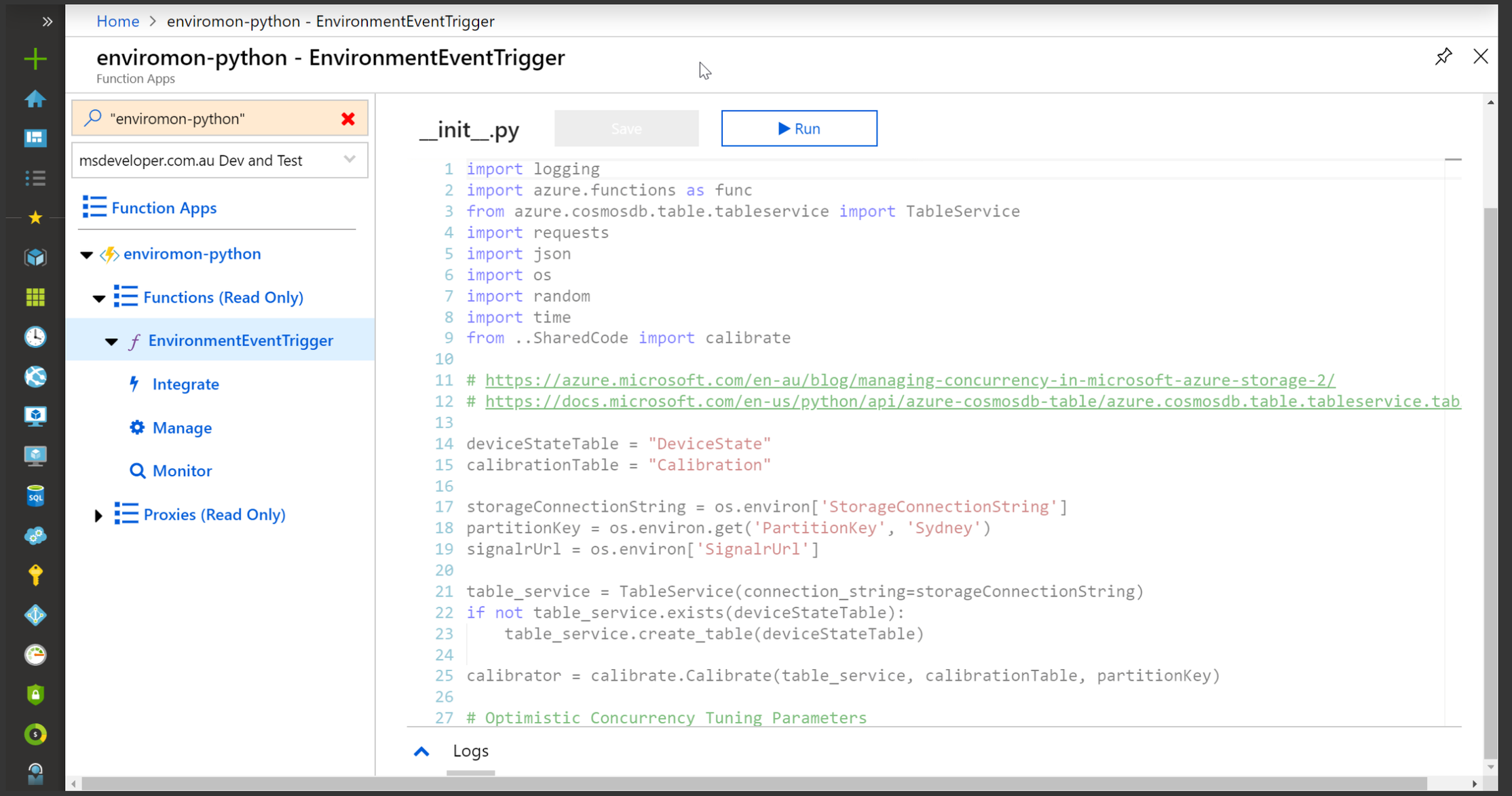
按需扩展

Scales on demand

运行成本低

Very **cheap** to run

# Python Azure Functions



# 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
MARKER
MARKER
MARKER
MARKER
MARKER
MARKER

Downloads/ | image-scanning-data - Jupyter N

localhost:8888/notebooks/Desktop/PyConDemos/image-scanning-data.ipynb#

jupyter image-scanning-data Last Checkpoint: an hour ago (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run | Code

```
Requirement already satisfied: cffi!=1.11.3,>=1.8 in c:\users\dglover\appdata\local\programs\python\python36\lib\site-pac
kages (from cryptography->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)
```

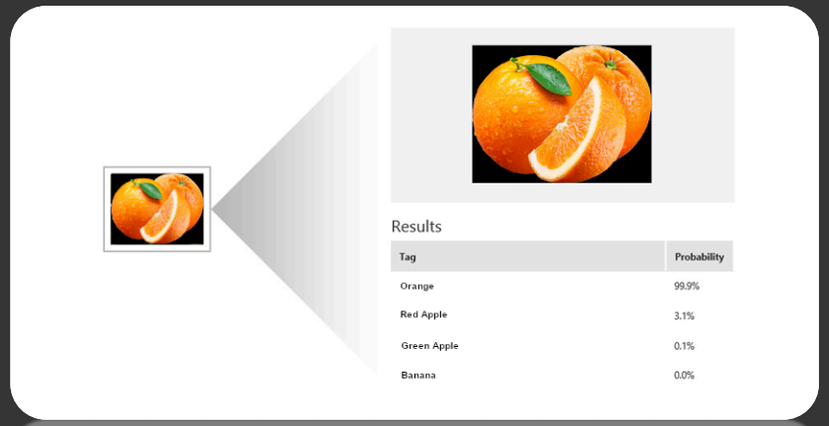
In [ ]:
```
x = []
y = []

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


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

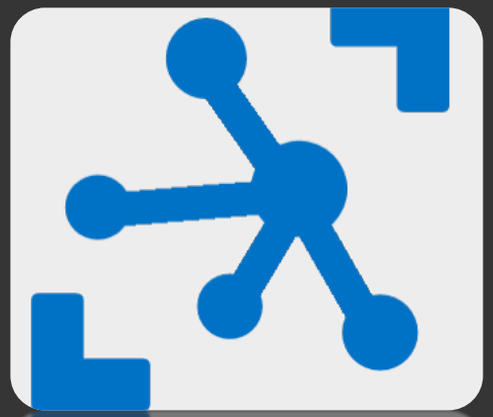
In [ ]:

In [ ]:

ENG 5:19 PM 18/09/2019

# Python端到端


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




jupyter

# 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

# 谢谢