Prospecta Software

#Calling Your First API Using Python

>>>import requests

>>requests.get(“<https://googleuser.me/api/>”)

<Response [200]>

#If you want to see the actual data,then you can use .text form the returned Response object:

>>>import requests

>>>response=requests.get(“<https://randomuser.me/api/>”)

>>>response.text

‘{“results”:[{“gender”:”male”, “name”:{“title”:”Mrs”,”first”:”Sai”,”last”:”T”}…

#Using the just-introduced TheDogAPI

>>>import requests

>>>response=requests.get(“<https://api.thedogapi.com/>”)

>>>response.text

‘{“message”:”the Dog API”)’

>>> response = requests.get("https://api.thedogapi.com/v1/breeds")

>>> response

<Response [200]>

>>> response.request

<PreparedRequest [GET]>

>>> request = response.request

>>> request.url

'https://api.thedogapi.com/v1/breeds'

>>> request.path\_url

'/v1/breeds'

>>> request.method

'GET'

>>> request.headers

{'User-Agent': 'python-requests/2.24.0', 'Accept-Encoding': 'gzip, deflate',

'Accept': '\*/\*', 'Connection': 'keep-alive'}

>>> response

<Response [200]>

>>> response.text

'[{"weight":{"imperial":"6 - 13","metric":"3 - 6"},

"height":{"imperial":"9 - 11.5","metric":"23 - 29"},"id":1,

"name":"Affenpinscher", ...}]'

>>> response.status\_code

200

>>> response.headers

{'Cache-Control': 'post-check=0, pre-check=0', 'Content-Encoding': 'gzip',

'Content-Type': 'application/json; charset=utf-8',

'Date': 'Sat, 25 Jul 2020 17:23:53 GMT'…}

#the status of a response using .status\_code and .reason

>>> response = requests.get("https://api.thedogapi.com/v1/breeds")

>>> response

<Response [200]>

>>> response.status\_code

200

>>> response.reason

'OK'

Q2:

#Request

GET [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001)?$select=name,revenue HTTP/1.1

Accept: application/json

Content-Type: application/json; charset=utf-8

OData-MaxVersion: 4.0

OData-Version: 4.0

#Response

HTTP/1.1 200 OK

Content-Type: application/json; odata.metadata=minimal

OData-Version: 4.0

{

"@odata.context": "[Organization URI]/api/data/v9.0/$metadata#accounts(name,revenue)/$entity",

"@odata.etag": "W/\"502186\"",

"name": "A. Datum Corporation (sample)",

"revenue": 10000,

"accountid": "00000000-0000-0000-0000-000000000001",

"\_transactioncurrencyid\_value":"b2a6b689-9a39-e611-80d2-00155db44581"

}

## #Retrieve a single property value

**#Request**

GET [Organization URI]/api/data/v9.0/accounts(00000000-0000-0000-0000-000000000001)/name HTTP/1.1

Accept: application/json

OData-MaxVersion: 4.0

OData-Version: 4.0

#Response

HTTP/1.1 200 OK

Content-Type: application/json; odata.metadata=minimal

OData-Version: 4.0

{

"@odata.context":"[Organization URI]/api/data/v9.0/$metadata#accounts(00000000-0000-0000-0000-000000000001)/name",

"value":"Adventure Works (sample)"

}

Q3:

### API security risks:

* **Broken object-level authorization.**
* **Broken function-level authorization.**
* **Broken user authentication.**
* **Excessive data exposure.**
* **Improper asset management.**
* **Improper asset management.**
* **Injection flaws.**
* **Mass assignment.**

### #API security best practices

#### Authenticate and authorize.

#### Implement access control.

#### Encrypt requests and responses.

#### Validate the data.

#### Assess your API risks.

#### Share only necessary information.

#### Choose your web services API.

#### Record APIs in an API registry.

#### Conduct regular security tests.

#### Stash your API keys.

#### Add AI to API monitoring and threat detection.

#### Understand the full scope of secure API consumption.