Face Recognition using Face API

In [3]:
```
import requests


headers = {
    'Content-Type': 'application/json',
    'Ocp-Apim-Subscription-Key': '38cb3720f312428aab6a2cd77217c4f8',
}
```

In [28]:
```
# Printing general information from the image
from pprint import pprint

body = dict()
body["url"] = "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQYjFsagXfnlNHE0YlmPbiYGsC6R8YRxsq2hg&usqp=C

body = str(body)

# Request URL
FaceApiDetect = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/detect?returnFaceId=true&returnFaceA

try:
    # REST Call
    response = requests.post(FaceApiDetect, data=body, headers=headers)
    pprint("RESPONSE:" + str(response.json()))

except Exception as e:
    print(e)
```

```
("RESPONSE:[{'faceId': 'c6405a5f-be60-4ed6-99ca-d9ab39e929f8', "
 "'faceRectangle': {'top': 32, 'left': 79, 'width': 36, 'height': 36}, "
 "'faceAttributes': {'smile': 1.0, 'headPose': {'pitch': -7.9, 'roll': 5.8, "
 "'yaw': 19.1}, 'gender': 'male', 'age': 37.0, 'facialHair': {'moustache': "
 "0.1, 'beard': 0.1, 'sideburns': 0.1}}}]")
```

In [115]:

```
personGroupId="friends"

body = dict()
body["name"] = "F.R.I.E.N.D.S"
body["userData"] = "All friends cast"
body = str(body)

#Request URL
FaceApiCreateLargePersonGroup = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/persongroups/'+persc

try:
    # REST Call
    response = requests.put(FaceApiCreateLargePersonGroup, data=body, headers=headers)
    print("RESPONSE:" + str(response.status_code))

except Exception as e:
    print(e)
```

```
RESPONSE:409
```

Chandler Image as Input

In [118]:

```
# For a single image
body = dict()
body["name"] = "Chandler"
body["userData"] = "Friends"
body = str(body)

#Request URL
FaceApiCreatePerson = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/persongroups/'+personGroupId+'

try:
    # REST Call
    response = requests.post(FaceApiCreatePerson, data=body, headers=headers)
    responseJson = response.json()
    personId = responseJson["personId"]
    print("PERSONID: "+str(personId))

except Exception as e:
    print(e)
```

```
PERSONID: b4eded72-debd-4228-b716-cfcc3573c95a
```

In [119]:
```
# For multiple images

chandlerImageList = ["https://i.pinimg.com/236x/b0/57/ff/b057ff0d16bd5205e4d3142e10f03394--muriel-matthew-perry.
                     "https://qph.fs.quoracdn.net/main-qimg-74677a162a39c79d6a9aa2b11cc195b1",
                     "https://pbs.twimg.com/profile_images/2991381736/e2160154f215a325b0fc73f866039311_400x400.j
                     "https://i.pinimg.com/236x/f2/9f/45/f29f45049768ddf5c5d75ff37ffbfb3f--hottest-actors-matthe
                    ]

#Request URL
FaceApiCreatePerson = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/persongroups/'+personGroupId+'

for image in chandlerImageList:
    body = dict()
    body["url"] = image
    body = str(body)

    try:
        # REST Call
        response = requests.post(FaceApiCreatePerson, data=body, headers=headers)
        responseJson = response.json()
        persistedFaceId = responseJson["persistedFaceId"]
        print("PERSISTED FACE ID: "+str(persistedFaceId))

    except Exception as e:
        print(e)
```

```
PERSISTED FACE ID: 46f6786b-faca-4c28-bd1f-f267572cde0b
PERSISTED FACE ID: 8820c4a3-42e6-425b-894b-4e2f6be1cce6
PERSISTED FACE ID: adce469d-a3a7-4312-9447-336349e2452a
PERSISTED FACE ID: c1b4047d-c512-473e-be48-2f5ea817f5d0
```

Ross images as input

In [122]:
```
# For a single input image

body = dict()
body["name"] = "ross"
body["userData"] = "Friends"
body = str(body)

#Request URL
FaceApiCreatePerson = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/persongroups/'+personGroupId+'

try:
    # REST Call
    response = requests.post(FaceApiCreatePerson, data=body, headers=headers)
    responseJson = response.json()
    personId = responseJson["personId"]
    print("PERSONID: "+str(personId))

except Exception as e:
    print(e)
```

```
PERSONID: e9fbff9e-0f17-4161-a36c-2950412aa5b7
```

In [123]:
```
# For multiple images

rossImageList = ["https://i.insider.com/5b33de335e48ec600b8b4580?width=1100&format=jpeg&auto=webp",
                 "https://a1cf74336522e87f135f-2f21ace9a6cf0052456644b80fa06d4f.ssl.cf2.rackcdn.com/images/ch
                 "https://upload.wikimedia.org/wikipedia/en/6/6f/David_Schwimmer_as_Ross_Geller.jpg",
                 "https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTf_FeCxUosWj_iZC2yf5b_GhJHQCUUpN9XIA&

#Request URL
FaceApiCreatePerson = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/persongroups/'+personGroupId+'

for image in rossImageList:
    body = dict()
    body["url"] = image
    body = str(body)

    try:
        # REST Call
        response = requests.post(FaceApiCreatePerson, data=body, headers=headers)
        responseJson = response.json()
        persistedFaceId = responseJson["persistedFaceId"]
        print("PERSISTED FACE ID: "+str(persistedFaceId))

    except Exception as e:
        print(e)
```

```
PERSISTED FACE ID: 7523d454-6e27-4878-8f29-57a039bdf311
PERSISTED FACE ID: 20707528-1ee1-4c8e-9622-c56d4248c578
PERSISTED FACE ID: 64402d15-4f1c-4028-9a6d-10c8ad872733
PERSISTED FACE ID: 98b44621-42fe-4f49-a8a9-7eb94c3480c9
```

To check training status,PersonGroupId is used since it stores every train image details — Get Training Status.

In [124]:
```
#Request Body
body = dict()

#Request URL
FaceApiTrain = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/persongroups/'+personGroupId+'/train'

try:
    # REST Call
    response = requests.post(FaceApiTrain, data=body, headers=headers)
    print("RESPONSE:" + str(response.status_code))

except Exception as e:
    print(e)
```

```
RESPONSE:202
```

Testing images on unknown data

Identify face image, this involves in total 3 different face APIs.

First API:

Face Detect(This will return a face Id which is valid for 24 hrs. This face id will be used by the next API to identify the character)

In [152]:

```
body = dict()
body["url"] = "https://gl-images.condecdn.net/image/jLpYBeXRj73/crop/405/f/Friends17_glamour_17apr13_NBCUniversa
body = str(body)

# Request URL
FaceApiDetect = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/detect?returnFaceId=true'

try:
    # REST Call
    response = requests.post(FaceApiDetect, data=body, headers=headers)
    responseJson = response.json()
    faceId = responseJson[0]["faceId"]
    print("FACE ID: "+str(faceId))

except Exception as e:
    print(e)
```

```
FACE ID: 33ee7061-3808-44f8-a437-b3efd71de89f
```

Second API: Face Identify(Face Identify will compute similarities between the query face and all the faces in the person group (given by personGroupId))

In [153]:
```
faceIdsList = [faceId]

body = dict()
body["personGroupId"] = personGroupId
body["faceIds"] = faceIdsList
body["maxNumOfCandidatesReturned"] = 1
body["confidenceThreshold"] = 0.5
body = str(body)

# Request URL
FaceApiIdentify = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/identify'

try:
    # REST Call
    response = requests.post(FaceApiIdentify, data=body, headers=headers)
    responseJson = response.json()
    personId = responseJson[0]["candidates"][0]["personId"]
    confidence = responseJson[0]["candidates"][0]["confidence"]
    print("PERSON ID: "+str(personId)+ ", CONFIDENCE :"+str(confidence))

except Exception as e:
    print("Could not detect")
```

```
PERSON ID: e9fbff9e-0f17-4161-a36c-2950412aa5b7, CONFIDENCE :0.87364
```

Third API: PersonGroup Person Details (Retrieve a person's name and userData)

In [154]:
```
# Request URL
FaceApiGetPerson = 'https://centralindia.api.cognitive.microsoft.com/face/v1.0/persongroups/'+personGroupId+'/pe

try:
    response = requests.get(FaceApiGetPerson, headers=headers)
    responseJson = response.json()
    print("This Is "+str(responseJson["name"]))

except Exception as e:
    print(e)
```

```
This Is ross
```

In [ ]: