

DRAG DROP -

You are developing a call to the Face API. The call must find similar faces from an existing list named `employeefaces`. The `employeefaces` list contains 60,000 images.

How should you complete the body of the HTTP request? To answer, drag the appropriate values to the correct targets. Each value may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:

Values**Answer Area**

```
{  
  "faceId": "18c51a87-3a69-47a8-aedc-a54745f708a1",  
  : "employeefaces",  
  "maxNumOfCandidatesReturned": 1,  
  "mode":   
}
```

Correct Answer:**Values****Answer Area**

```
{  
  "faceId": "18c51a87-3a69-47a8-aedc-a54745f708a1",  
  "LargeFaceListId": "employeefaces",  
  "maxNumOfCandidatesReturned": 1,  
  "mode": "matchFace"  
}
```

Box 1: LargeFaceListID -

LargeFaceList: Add a face to a specified large face list, up to 1,000,000 faces.

Note: Given query face's `faceId`, to search the similar-looking faces from a `faceId` array, a face list or a large face list. A `"faceListId"` is created by FaceList - Create containing persisted `faceIds` that will not expire. And a `"largeFaceListId"` is created by LargeFaceList - Create containing persisted `faceIds` that will also not expire.

Incorrect Answers:

Not `"faceListId"`: Add a face to a specified face list, up to 1,000 faces.

Box 2: matchFace -

Find similar has two working modes, `"matchPerson"` and `"matchFace"`. `"matchPerson"` is the default mode that it tries to find faces of the same person as possible by using internal same-person thresholds. It is useful to find a known person's other photos. Note that an empty list will be returned if no faces pass the internal thresholds. `"matchFace"` mode ignores same-person thresholds and returns ranked similar faces anyway, even the similarity is low. It can be used in the cases like searching celebrity-looking faces.

Reference:

<https://docs.microsoft.com/en-us/rest/api/faceapi/face/findsimilar>

DRAG DROP -

You are developing a photo application that will find photos of a person based on a sample image by using the Face API.

You need to create a POST request to find the photos.

How should you complete the request? To answer, drag the appropriate values to the correct targets. Each value may be used once, more than once, or not at all.

You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:

Values

detect

findsimilars

group

identify

matchFace

matchPerson

verify

Answer Area

POST {Endpoint}/face/v1.0/

Request Body

{
 "faceId": "c5c24a82-6845-4031-9d5d-978df9175426",
 "largeFaceListId": "sample_list",
 "maxNumOfCandidatesReturned": 10,
 "mode": ""
}

Correct Answer:

Values

detect

findsimilars

group

identify

matchFace

matchPerson

verify

Answer Area

POST {Endpoint}/face/v1.0/

detect

Request Body

{
 "faceId": "c5c24a82-6845-4031-9d5d-978df9175426",
 "largeFaceListId": "sample_list",
 "maxNumOfCandidatesReturned": 10,
 "mode": "

matchPerson

"
}

Box 1: detect -

Face - Detect With Url: Detect human faces in an image, return face rectangles, and optionally with faceIds, landmarks, and attributes.

POST {Endpoint}/face/v1.0/detect

Box 2: matchPerson -

Find similar has two working modes, "matchPerson" and "matchFace". "matchPerson" is the default mode that it tries to find faces of the same person as possible by using internal same-person thresholds. It is useful to find a known person's other photos. Note that an empty list will be returned if no faces pass the internal thresholds. "matchFace" mode ignores same-person thresholds and returns ranked similar faces anyway, even the similarity is low. It can be used in the cases like searching celebrity-looking faces.

Reference:

<https://docs.microsoft.com/en-us/rest/api/faceapi/face/detectwithurl> <https://docs.microsoft.com/en-us/rest/api/faceapi/face/findsimilar>

HOTSPOT -

You develop a test method to verify the results retrieved from a call to the Computer Vision API. The call is used to analyze the existence of company logos in images. The call returns a collection of brands named brands.

You have the following code segment.

```
for brand in image_analysis.brands:
    if brand_confidence >= 0.75:
        print(f"\nLogo of {brand_name} between {brand.rectangle_x}, {brand.rectangle.y} and
{brand.rectangle.w}, {brand.rectangle.h}")
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Statements	Yes	No
The code will return the name of each detected brand with a confidence equal to or higher than 75 percent.	<input type="radio"/>	<input type="radio"/>
The code will return coordinates for the top-left corner of the rectangle that contains the brand logo of the displayed brands.	<input type="radio"/>	<input type="radio"/>
The code will return coordinates for the bottom-right corner of the rectangle that contains the brand logo of the displayed brands.	<input type="radio"/>	<input type="radio"/>

Correct Answer:

Answer Area

Statements	Yes	No
The code will return the name of each detected brand with a confidence equal to or higher than 75 percent.	<input checked="" type="radio"/>	<input type="radio"/>
The code will return coordinates for the top-left corner of the rectangle that contains the brand logo of the displayed brands.	<input checked="" type="radio"/>	<input type="radio"/>
The code will return coordinates for the bottom-right corner of the rectangle that contains the brand logo of the displayed brands.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: Yes -

Box 2: Yes -

Coordinates of a rectangle in the API refer to the top left corner.

Box 3: No -

Reference:

<https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/concept-brand-detection>

HOTSPOT -

You develop an application that uses the Face API.

You need to add multiple images to a person group.

How should you complete the code? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

```
Parallel.For(0, PersonCount, async i =>
{
    Guid personId = persons[i].PersonId;
    string personImageDir = $"/path/to/person/{i}/images";
    foreach (string imagePath in Directory.GetFiles(personImageDir, "*.jpg"))
    {
        using ( ▼ t = File.OpenRead(imagePath))
            

|        |
|--------|
| File   |
| Stream |
| Uri    |
| Url    |


        {
            await faceClient.PersonGroupPerson. ▼

|                        |
|------------------------|
| AddFaceFromStreamAsync |
| AddFaceFromUrlAsync    |
| CreateAsync            |
| GetAsync               |


            (personGroupId, personId, t);
        }
    }
});
```

Correct Answer:

Answer Area

```
Parallel.For(0, PersonCount, async i =>
{
    Guid personId = persons[i].PersonId;
    string personImageDir = $"/path/to/person/{i}/images";
    foreach (string imagePath in Directory.GetFiles(personImageDir, "*.jpg"))
    {
        using ( ▼ t = File.OpenRead(imagePath))
            

|        |
|--------|
| File   |
| Stream |
| Uri    |
| Url    |


        {
            await faceClient.PersonGroupPerson. ▼

|                        |
|------------------------|
| AddFaceFromStreamAsync |
| AddFaceFromUrlAsync    |
| CreateAsync            |
| GetAsync               |


            (personGroupId, personId, t);
        }
    }
});
```

Box 1: Stream -

The File.OpenRead(String) method opens an existing file for reading.

Example: Open the stream and read it back.

```
using (FileStream fs = File.OpenRead(path))
```

Box 2: CreateAsync -

Create the persons for the PersonGroup. Persons are created concurrently.

Example:

```
await faceClient.PersonGroupPerson.CreateAsync(personGroupId, personName);
```

Reference:

<https://docs.microsoft.com/en-us/azure/cognitive-services/face/face-api-how-to-topics/how-to-add-faces>

Question #16

Topic 2

Your company uses an Azure Cognitive Services solution to detect faces in uploaded images. The method to detect the faces uses the following code.

```
static async Task DetectFaces(string imagePath)
{
    HttpClient client = new HttpClient();
    DefaultRequestHeaders.Add("Ocp-Apim-Subscription-Key", subscriptionKey);
    string requestParameter = "detectionModel=detection_01&returnFaceId=true&returnFaceLandmarks=false";
    string uri = endpoint + "/face/v1.0/detect?" + requestParameters;
    HttpResponseMessage response;
    byte[] byteData = GetImagesAsByteArray(imageFilePath);
    using (ByteArrayContent content = new ByteArrayContent(byteData))
    {
        Headers.ContentType = new MediaTypeHeaderValue("application/octet-stream");
        response = await PostAsync(uri, content);
        string contentString = await Content.ReadAsStringAsync();
        ProcessDetection(contentString);
    }
}
```

You discover that the solution frequently fails to detect faces in blurred images and in images that contain sideways faces.

You need to increase the likelihood that the solution can detect faces in blurred images and images that contain sideways faces.

What should you do?

- A. Use a different version of the Face API.
- B. Use the Computer Vision service instead of the Face service.
- C. Use the Identify method instead of the Detect method.

D. Change the detection model. **Most Voted**

Correct Answer: D

Community vote distribution

D (100%)

You have the following Python function for creating Azure Cognitive Services resources programmatically. `def create_resource(resource_name, kind, account_tier, location) : parameters = CognitiveServicesAccount(sku=Sku(name=account_tier), kind=kind, location=location, properties={}) result = client.accounts.create(resource_group_name, resource_name, parameters)`

You need to call the function to create a free Azure resource in the West US Azure region. The resource will be used to generate captions of images automatically.

Which code should you use?

- A. `create_resource("res1", "ComputerVision", "F0", "westus")` **Most Voted**
- B. `create_resource("res1", "CustomVision.Prediction", "F0", "westus")`
- C. `create_resource("res1", "ComputerVision", "S0", "westus")`
- D. `create_resource("res1", "CustomVision.Prediction", "S0", "westus")`

Correct Answer: A

Community vote distribution

A (98%)

You are developing a method that uses the Computer Vision client library. The method will perform optical character recognition (OCR) in images. The method has the following code.

```
def read_file_url(computervision_client, url_file):
    read_response = computervision_client.read(url_file, raw=True)
    read_operation_location = read_response.headers["Operation-Location"]
    operation_id = read_operation_location.split("/")[-1]
    read_result = computervision_client.get_read_result(operation_id)

    for page in read_result.analyze_result.read_results:
        for line in page.lines:
            print(line.text)
```

During testing, you discover that the call to the `GetReadResultAsync` method occurs before the read operation is complete.

You need to prevent the `GetReadResultAsync` method from proceeding until the read operation is complete.

Which two actions should you perform? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Remove the `operation_id` parameter.
- B. Add code to verify the `read_results.status` value. **Most Voted**
- C. Add code to verify the status of the `read_operation_location` value.
- D. Wrap the call to `get_read_result` within a loop that contains a delay. **Most Voted**

Correct Answer: BD

Community vote distribution

BD (100%)

HOTSPOT -

You are building an app that will enable users to upload images. The solution must meet the following requirements:

- * Automatically suggest alt text for the images.
- * Detect inappropriate images and block them.
- * Minimize development effort.

You need to recommend a computer vision endpoint for each requirement.

What should you recommend? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Generate alt text:

https://westus.api.cognitive.microsoft.com/contentmoderator/moderate/v1.0/ProcessImage/Evaluate
https://westus.api.cognitive.microsoft.com/customvision/v3.1/prediction/projectId/classify/iterations/publishedName/image
https://westus.api.cognitive.microsoft.com/vision/v3.2/analyze/?visualFeatures=Adult,Descnption

Detect inappropriate content:

https://westus.api.cognitive.microsoft.com/contentmoderator/moderate/v1.0/ProcessImage/Evaluate
https://westus.api.cognitive.microsoft.com/customvision/v3.1/prediction/projectId/classify/iterations/publishedName/image
https://westus.api.cognitive.microsoft.com/vision/v3.2/analyze/?visualFeatures=Adult,Description
https://westus.api.cognitive.microsoft.com/vision/v3.2/describe?maxCandidates=1

Correct Answer:

Answer Area

Generate alt text:

https://westus.api.cognitive.microsoft.com/contentmoderator/moderate/v1.0/ProcessImage/Evaluate
https://westus.api.cognitive.microsoft.com/customvision/v3.1/prediction/projectId/classify/iterations/publishedName/image
https://westus.api.cognitive.microsoft.com/vision/v3.2/analyze/?visualFeatures=Adult,Descnption

Detect inappropriate content:

https://westus.api.cognitive.microsoft.com/contentmoderator/moderate/v1.0/ProcessImage/Evaluate
https://westus.api.cognitive.microsoft.com/customvision/v3.1/prediction/projectId/classify/iterations/publishedName/image
https://westus.api.cognitive.microsoft.com/vision/v3.2/analyze/?visualFeatures=Adult,Description
https://westus.api.cognitive.microsoft.com/vision/v3.2/describe?maxCandidates=1

Box 1: <https://westus.api.cognitive.microsoft.com/customvision/v3.1/prediction/projectid/classify/iterations/publishName/image>

Box 2: <https://westus.api.cognitive.microsoft.com/vision/v3.2/analyze/?visualFeatures=Adult,Description>

Computer Vision can detect adult material in images so that developers can restrict the display of these images in their software. Content flags are applied with a score between zero and one so developers can interpret the results according to their own preferences.

You can detect adult content with the Analyze Image API. When you add the value of Adult to the visualFeatures query parameter

Incorrect:

Use the Image Moderation API in Azure Content Moderator to scan image content. The moderation job scans your content for profanity, and compares it against custom and shared blocklists.

Reference:

<https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/concept-detecting-adult-content> <https://docs.microsoft.com/en-us/azure/cognitive-services/content-moderator/try-image-api> <https://docs.microsoft.com/en-us/legal/cognitive-services/custom-vision/custom-vision-cvs-transparency-note>

You need to build a solution that will use optical character recognition (OCR) to scan sensitive documents by using the Computer Vision API. The solution must NOT be deployed to the public cloud. What should you do?

- A. Build an on-premises web app to query the Computer Vision endpoint.
- B. Host the Computer Vision endpoint in a container on an on-premises server. Most Voted
- C. Host an exported Open Neural Network Exchange (ONNX) model on an on-premises server.
- D. Build an Azure web app to query the Computer Vision endpoint.

Correct Answer: B

Community vote distribution

B (100%)

You have an Azure Cognitive Search solution and a collection of handwritten letters stored as JPEG files. You plan to index the collection. The solution must ensure that queries can be performed on the contents of the letters. You need to create an indexer that has a skillset. Which skill should you include?

- A. image analysis
- B. optical character recognition (OCR) Most Voted
- C. key phrase extraction
- D. document extraction

Correct Answer: B

Community vote distribution

B (95%)

5%

HOTSPOT

You have a library that contains thousands of images.

You need to tag the images as photographs, drawings, or clipart.

Which service endpoint and response property should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Answer Area

Service endpoint:

- Computer Vision analyze images
- Computer Vision object detection
- Custom Vision image classification
- Custom Vision object detection

Property:

- categories
- description
- imageType
- metadata
- objects

Correct Answer:

Answer Area

Service endpoint:

- Computer Vision analyze images
- Computer Vision object detection
- Custom Vision image classification
- Custom Vision object detection

Property:

- categories
- description
- imageType
- metadata
- objects

You have an app that captures live video of exam candidates.

You need to use the Face service to validate that the subjects of the videos are real people.

What should you do?

- A. Call the face detection API and retrieve the face rectangle by using the FaceRectangle attribute.
- B. Call the face detection API repeatedly and check for changes to the FaceAttributes.HeadPose attribute. Most Voted
- C. Call the face detection API and use the FaceLandmarks attribute to calculate the distance between pupils.
- D. Call the face detection API repeatedly and check for changes to the FaceAttributes.Accessories attribute.

Correct Answer: B

Community vote distribution

B (88%)

12%

题库来源阿泽Vx: est258258