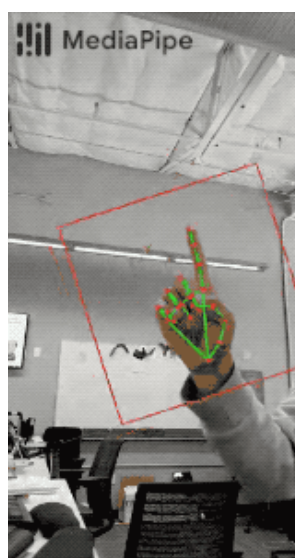
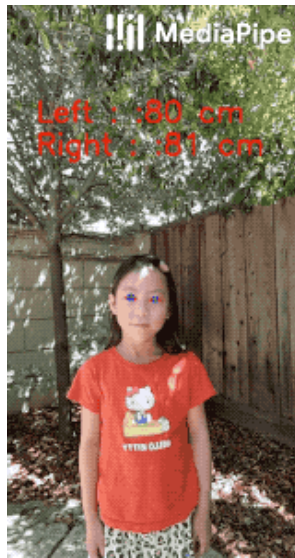


실습 모델 소개 : MediaPipe

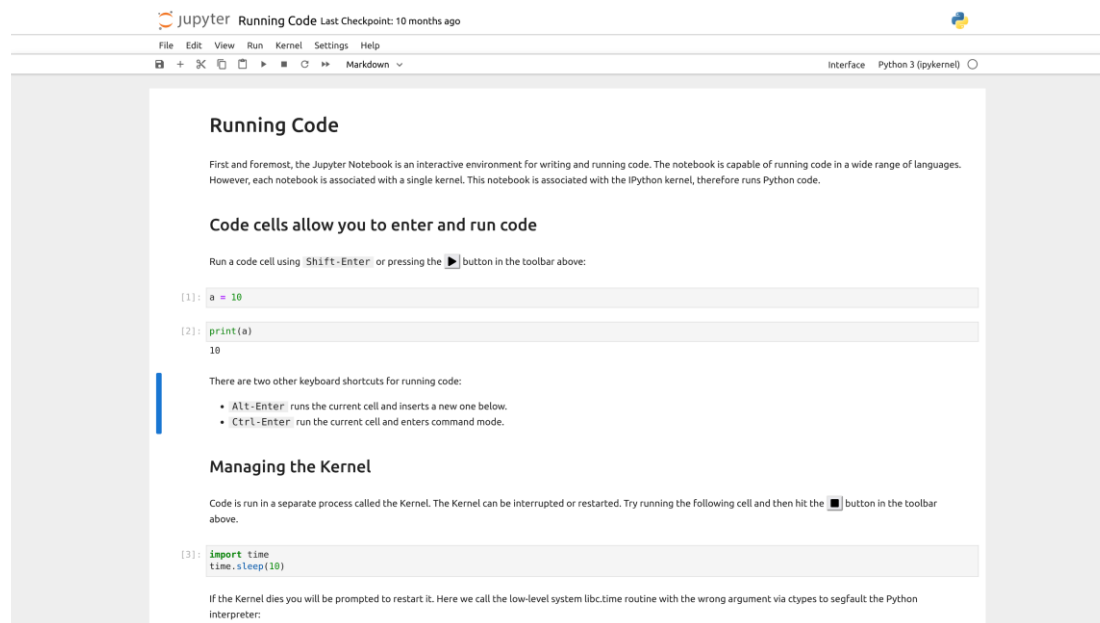


구글에서 2019년 개발된, 인체를 대상으로 하는 비전 인식 기능들을 학습이 완료된 상태로 제공하는 서비스

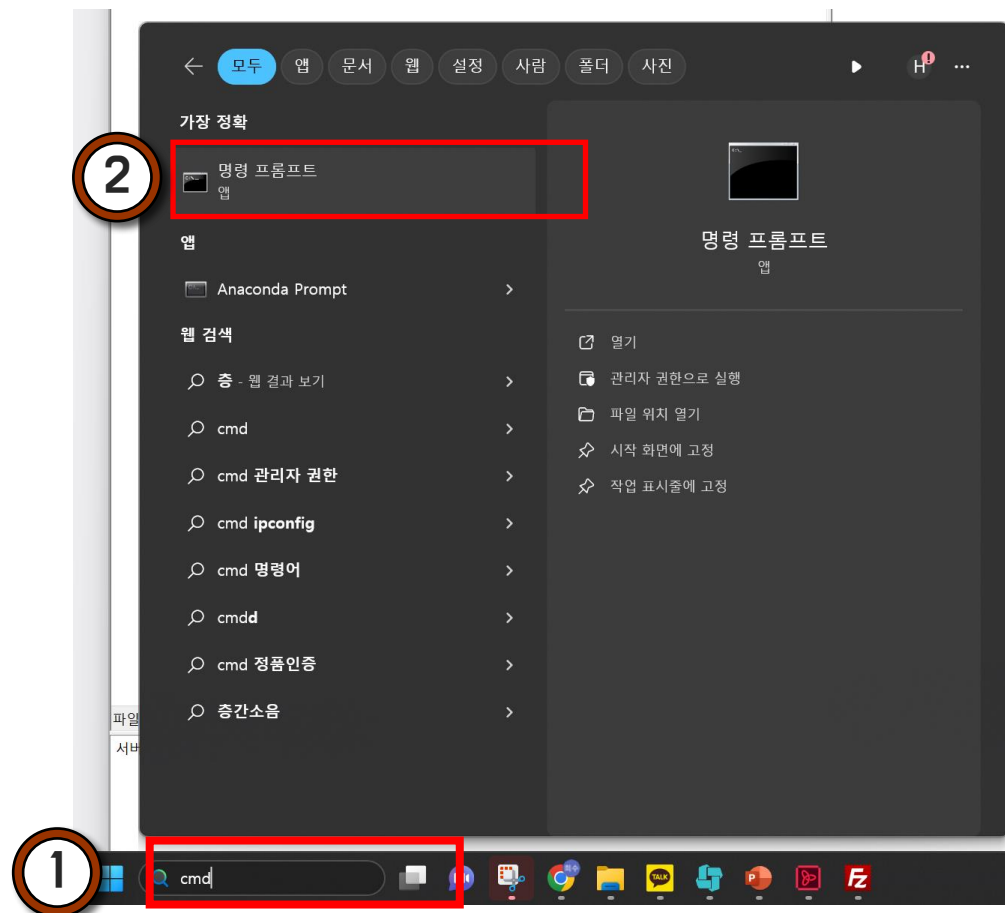


실습 환경 소개 : Jupyter Notebook

Jupyter에서 제작한 파이썬용 통합 개발 환경
셀 단위로 나누어져 있으며, 셀 단위로 실행 결과를 볼 수 있어 직관적인 개발 툴



Jupyter Notebook 실행하기









Jupyter Notebook 실행하기 - 성공!

[Quit](#)[Logout](#)[Files](#)[Running](#)[Clusters](#)

Select items to perform actions on them.

[Upload](#)[New ▾](#)

| <input type="checkbox"/> 0 ▾ |  / | Name ▾ | Last Modified | File size |
|------------------------------|---|--------|---------------|-----------|
| <input type="checkbox"/> |  코드_1일차_MediaPipe.ipynb | | 7일 전 | 157 kB |
| <input type="checkbox"/> |  코드_1일차_mediapipe_facemash.ipynb | | 7일 전 | 7.15 kB |
| <input type="checkbox"/> |  코드_1일차_이미지분류.ipynb | | 7일 전 | 845 kB |
| <input type="checkbox"/> |  cat.jpg | | 7달 전 | 75.4 kB |
| <input type="checkbox"/> |  imagenet_classes.txt | | 10달 전 | 16.4 kB |

◈ 코드_1일차_MediaPipe.ipynb

Code

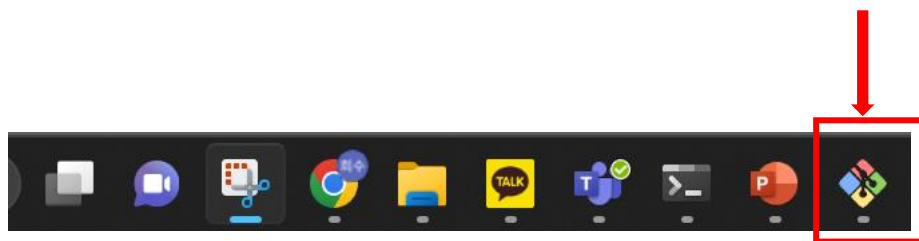
```
!pip install opencv-python
```

Code

```
!pip install mediapipe
```

필요한 모듈 설치

코드_1일차_MediaPipe.ipynb



main 코드 실행 후 잠시 기다리면
하단 작업표시줄에 실행창(카메라 창)이 생성됨
클릭하여 포즈 인식을 진행해보세요!

Code

```
import cv2
import mediapipe as mp

mp_drawing = mp.solutions.drawing_utils
mp_hands = mp.solutions.hands

cap = cv2.VideoCapture(0)
with mp_hands.Hands(
    max_num_hands=1,
    min_detection_confidence=0.5,
    min_tracking_confidence=0.5) as hands:

    while cap.isOpened():
        success, image = cap.read()
        if not success:
            continue
        image = cv2.cvtColor(cv2.flip(image, 1), cv2.COLOR_BGR2RGB)

        results = hands.process(image)

        image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)

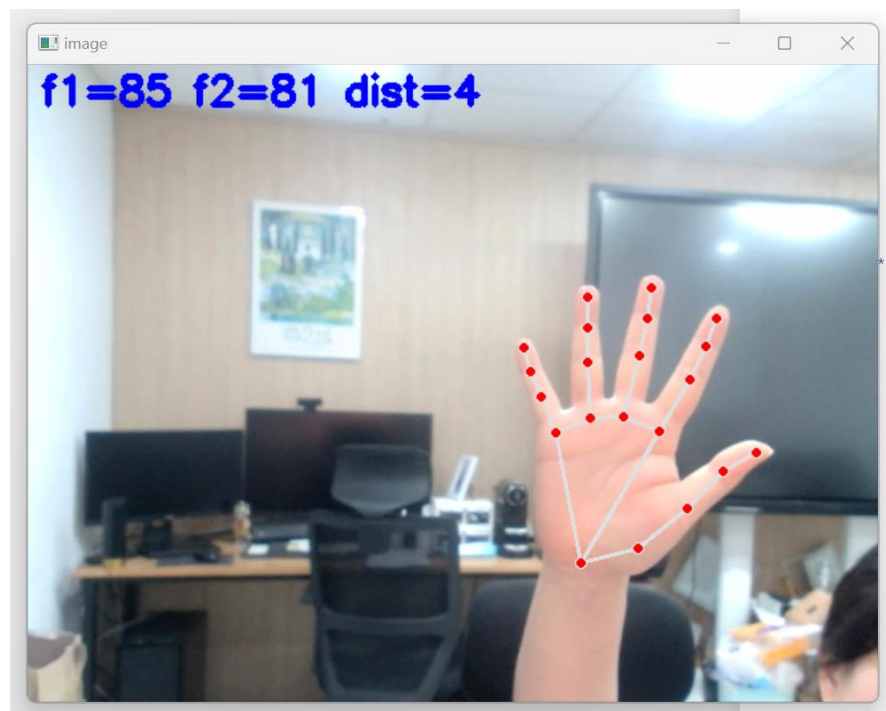
        if results.multi_hand_landmarks:
            for hand_landmarks in results.multi_hand_landmarks:
                finger1 = int(hand_landmarks.landmark[4].x * 100)
                finger2 = int(hand_landmarks.landmark[8].x * 100)
                dist = abs(finger1 - finger2)
                cv2.putText(
                    image, text='f1=%d f2=%d dist=%d' % (finger1, finger2, dist), org=(10, 30),
                    fontFace=cv2.FONT_HERSHEY_SIMPLEX, fontScale=1,
                    color=255, thickness=3)

                mp_drawing.draw_landmarks(
                    image, hand_landmarks, mp_hands.HAND_CONNECTIONS)

            cv2.imshow('image', image)
            if cv2.waitKey(1) == ord('q'):
                break

    cap.release()
    cv2.destroyAllWindows()
```


◈ 코드_1일차_MediaPipe.ipynb



생성된 카메라창을 열어주고 손을 카메라에 비치면 손의 뼈대를 인식함
카메라를 종료하려면 키보드 q버튼 누르기!

◈ 코드_1일차_MediaPipe.ipynb

Code

```
import cv2
import mediapipe as mp

mp_drawing = mp.solutions.drawing_utils
mp_hands = mp.solutions.hands

cap = cv2.VideoCapture(0)
with mp_hands.Hands(
    max_num_hands=1,
    ...
```

인식할 손의 개수를 지정하는 코드
숫자를 바꿔가면서 여러 손을 인식해보세요!

◆ 코드_1일차_MediaPipe_facemash.ipynb

Code

```
!pip install opencv-python
```

Code

```
!pip install mediapipe
```

Code

```
import cv2
import mediapipe as mp
mp_drawing = mp.solutions.drawing_utils
mp_drawing_styles =
mp.solutions.drawing_styles
mp_face_mesh = mp.solutions.face_mesh
```

필요한 모듈 및 변수 세팅



코드_1일차_MediaPipe_facemash.ipynb

적용할 스타일 :

main 코드 실행 후 잠시 기다리면 코드 하단에 입력창이 생성됨
얼굴을 인식하는데 적용할 스타일을 작성하는 칸
1~3까지의 숫자를 조합하여 적어보세요!

ex) 1, 12, 13, 23, 123,...

Code

```
drawing_spec = mp_drawing.DrawingSpec(thickness=1, circle_radius=1)
cap = cv2.VideoCapture(0)
with mp_face_mesh.FaceMesh(
    max_num_faces=1,
    refine_landmarks=True,
    min_detection_confidence=0.5,
    min_tracking_confidence=0.5) as face_mesh:
    num = input('적용할 스타일: ')

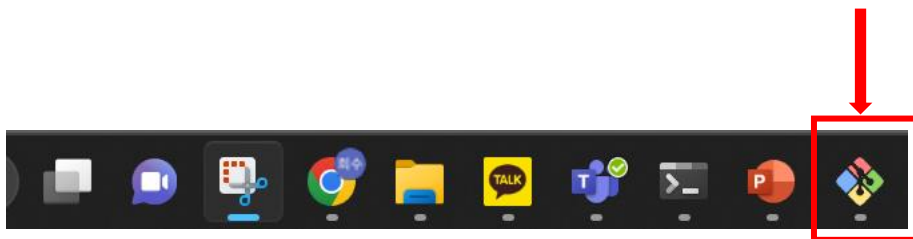
    while cap.isOpened():
        success, image = cap.read()
        if not success:
            print("Ignoring empty camera frame.")
            continue

        image.flags.writeable = False
        image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
        results = face_mesh.process(image)
        image.flags.writeable = True
        image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
        if results.multi_face_landmarks:
            for face_landmarks in results.multi_face_landmarks:
                if '1' in num:
                    mp_drawing.draw_landmarks(
                        image=image,
                        landmark_list=face_landmarks,
                        connections=mp_face_mesh.FACEMESH_TESSELATION,
                        landmark_drawing_spec=None,
                        connection_drawing_spec=mp_drawing_styles
                            .get_default_face_mesh_tessellation_style())
                if '2' in num:
                    mp_drawing.draw_landmarks(
                        image=image,
                        landmark_list=face_landmarks,
                        connections=mp_face_mesh.FACEMESH_CONTOURS,
                        landmark_drawing_spec=None,
                        connection_drawing_spec=mp_drawing_styles
                            .get_default_face_mesh_contours_style())
                if '3' in num:
                    mp_drawing.draw_landmarks(
                        image=image,
                        landmark_list=face_landmarks,
                        connections=mp_face_mesh.FACEMESH_IRISES,
                        landmark_drawing_spec=None,
                        connection_drawing_spec=mp_drawing_styles
                            .get_default_face_mesh_iris_connections_style())

        cv2.imshow('MediaPipe Face Mesh', cv2.flip(image, 1))
        if cv2.waitKey(1) == ord('q'):
            break

    cap.release()
    cv2.destroyAllWindows()
```

코드_1일차_MediaPipe_facemash.ipynb



적용할 스타일을 입력하면 작업표시줄에 카메라 실행창이 생성됨
클릭하여 포즈 인식을 진행해보세요!

Code

```
drawing_spec = mp_drawing.DrawingSpec(thickness=1, circle_radius=1)
cap = cv2.VideoCapture(0)
with mp_face_mesh.FaceMesh(
    max_num_faces=1,
    refine_landmarks=True,
    min_detection_confidence=0.5,
    min_tracking_confidence=0.5) as face_mesh:
    num = input('적용할 스타일: ')

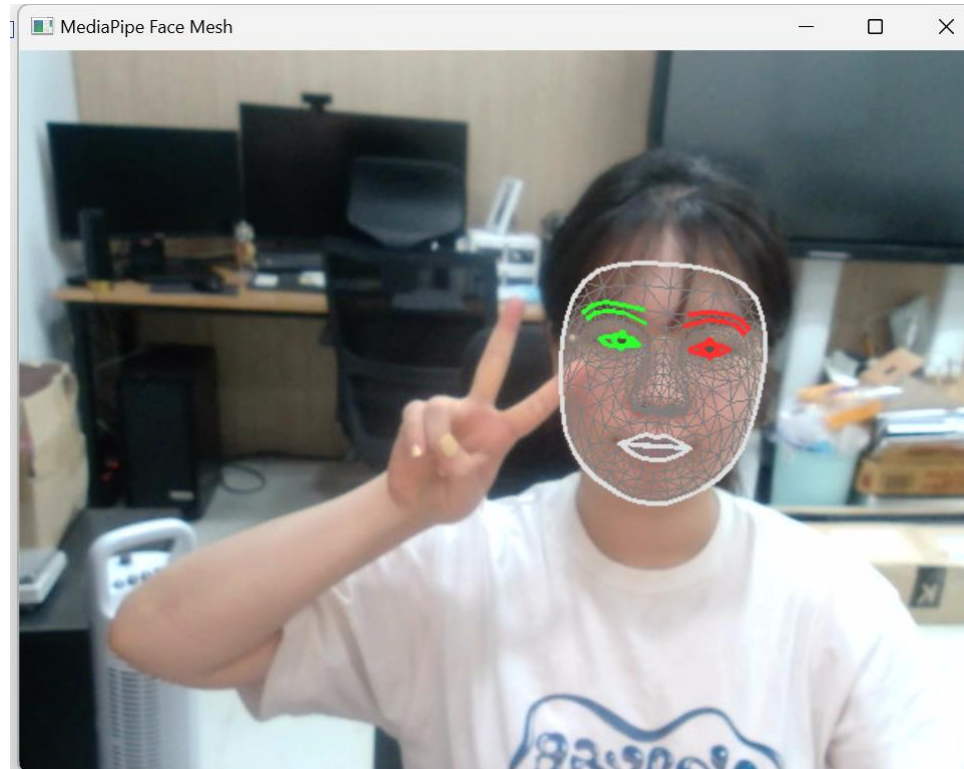
    while cap.isOpened():
        success, image = cap.read()
        if not success:
            print("Ignoring empty camera frame.")
            continue

        image.flags.writeable = False
        image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
        results = face_mesh.process(image)
        image.flags.writeable = True
        image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
        if results.multi_face_landmarks:
            for face_landmarks in results.multi_face_landmarks:
                if '1' in num:
                    mp_drawing.draw_landmarks(
                        image=image,
                        landmark_list=face_landmarks,
                        connections=mp_face_mesh.FACEMESH_TESSELATION,
                        landmark_drawing_spec=None,
                        connection_drawing_spec=mp_drawing_styles
                            .get_default_face_mesh_tessellation_style())
                if '2' in num:
                    mp_drawing.draw_landmarks(
                        image=image,
                        landmark_list=face_landmarks,
                        connections=mp_face_mesh.FACEMESH_CONTOURS,
                        landmark_drawing_spec=None,
                        connection_drawing_spec=mp_drawing_styles
                            .get_default_face_mesh_contours_style())
                if '3' in num:
                    mp_drawing.draw_landmarks(
                        image=image,
                        landmark_list=face_landmarks,
                        connections=mp_face_mesh.FACEMESH_IRISES,
                        landmark_drawing_spec=None,
                        connection_drawing_spec=mp_drawing_styles
                            .get_default_face_mesh_iris_connections_style())

        cv2.imshow('MediaPipe Face Mesh', cv2.flip(image, 1))
        if cv2.waitKey(1) == ord('q'):
            break

    cap.release()
    cv2.destroyAllWindows()
```

◈ 코드_1일차_MediaPipe_facemash.ipynb



스타일 123 적용 예시

생성된 카메라창을 열어주고 얼굴을 카메라에 비치면 얼굴을 인식함
카메라를 종료하려면 키보드 q버튼 누르기!