```
!pip install ultralytics  # 라이브러리 설치
```

```
Successfully installed nvidia-cublas-cu12-12.1.3.1 nvidia-cuda-cupti-cu12-12.1.105 nvidia-cuda-nvrtc-cu12-12.1.105 nvidia-cuda-runtime-cu12-12.1.105 n
```

```
# 라이브러리 불러오기
from ultralytics import YOLO # YOLO v8 Quick Manual


# 모델 선언 모델 구조 및 가중치 설정
model = YOLO()
```

```
Downloading https://github.com/ultralytics/assets/releases/download/v8.1.0/yolov8n.pt to 'yolov8n.pt'...
100%|██████████| 6.23M/6.23M [00:00<00:00, 110MB/s]
```

```
# 모델 학습 학습에 관련된 설정 가능
model.train()
```

```
      98/100         0G     0.4467     0.5005     0.9204         13        640: 100%|██████████| 1/1 [00:03<00:00,  3.03s/it]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:01<00:00,  1.15s/it]


      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
      99/100         0G     0.6337     0.6338      1.133         13        640: 100%|██████████| 1/1 [00:03<00:00,  3.01s/it]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:01<00:00,  1.13s/it]


      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
     100/100         0G     0.6111     0.7242      1.069         13        640: 100%|██████████| 1/1 [00:03<00:00,  3.86s/it]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:01<00:00,  1.14s/it]


100 epochs completed in 0.137 hours.
Optimizer stripped from runs/detect/train/weights/last.pt, 6.6MB
Optimizer stripped from runs/detect/train/weights/best.pt, 6.5MB

Validating runs/detect/train/weights/best.pt...
Ultralytics YOLOv8.1.47 🚀 Python-3.10.12 torch-2.2.1+cu121 CPU (Intel Xeon 2.20GHz)
Model summary (fused): 168 layers, 3151904 parameters, 0 gradients, 8.7 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:00<00:00,  1.02it/s]
                   all          4         17      0.541      0.859      0.873      0.621
                person          4         10      0.639        0.5       0.51      0.285
                   dog          4          1      0.316          1      0.995      0.597
                 horse          4          2      0.627          1      0.995      0.648
              elephant          4          2      0.385      0.656      0.745      0.303
              umbrella          4          1      0.536          1      0.995      0.995
          potted plant          4          1      0.741          1      0.995      0.895
Speed: 2.3ms preprocess, 226.9ms inference, 0.0ms loss, 1.9ms postprocess per image
Results saved to runs/detect/train
ultralytics.utils.metrics.DetMetrics object with attributes:

ap_class_index: array([ 0, 16, 17, 20, 25, 58])
box: ultralytics.utils.metrics.Metric object
confusion_matrix: <ultralytics.utils.metrics.ConfusionMatrix object at 0x7a95d2dd6d70>
curves: ['Precision-Recall(B)', 'F1-Confidence(B)', 'Precision-Confidence(B)', 'Recall-Confidence(B)']
curves_results: [[array([          0,    0.001001,    0.002002,    0.003003,    0.004004,    0.005005,    0.006006,    0.007007,    0.008008,
0.009009,     0.01001,    0.011011,    0.012012,    0.013013,    0.014014,    0.015015,    0.016016,    0.017017,    0.018018,    0.019019,
0.02002,    0.021021,    0.022022,    0.023023,
          0.024024,    0.025025,    0.026026,    0.027027,    0.028028,    0.029029,     0.03003,    0.031031,    0.032032,    0.033033,
0.034034,    0.035035,    0.036036,    0.037037,    0.038038,    0.039039,     0.04004,    0.041041,    0.042042,    0.043043,    0.044044,
0.045045,    0.046046,    0.047047,
          0.048048,    0.049049,     0.05005,    0.051051,    0.052052,    0.053053,    0.054054,    0.055055,    0.056056,    0.057057,
0.058058,    0.059059,     0.06006,    0.061061,    0.062062,    0.063063,    0.064064,    0.065065,    0.066066,    0.067067,    0.068068,
0.069069,     0.07007,    0.071071,
          0.072072,    0.073073,    0.074074,    0.075075,    0.076076,    0.077077,    0.078078,    0.079079,     0.08008,    0.081081,
0.082082,    0.083083,    0.084084,    0.085085,    0.086086,    0.087087,    0.088088,    0.089089,     0.09009,    0.091091,    0.092092,
0.093093,    0.094094,    0.095095,
          0.096096,    0.097097,    0.098098,    0.099099,      0.1001,      0.1011,      0.1021,      0.1031,      0.1041,     0.10511,
0.10611,     0.10711,     0.10811,     0.10911,     0.11011,     0.11111,     0.11211,     0.11311,     0.11411,     0.11512,     0.11612,
0.11712,     0.11812,     0.11912,
           0.12012,     0.12112,     0.12212,     0.12312,     0.12412,     0.12513,     0.12613,     0.12713,     0.12813,     0.12913,
0.13013,     0.13113,     0.13213,     0.13313,     0.13413,     0.13514,     0.13614,     0.13714,     0.13814,     0.13914,     0.14014,
0.14114,     0.14214,     0.14314,
           0.14414,     0.14515,     0.14615,     0.14715,     0.14815,     0.14915,     0.15015,     0.15115,     0.15215,     0.15315,
0.15415,     0.15516,     0.15616,     0.15716,     0.15816,     0.15916,     0.16016,     0.16116,     0.16216,     0.16316,     0.16416,
```

```
# model.val()
```

```
# 예측값 생성 데이터의 예측 결과 생성
model.predict(save=True, save_txt=True)
```

```
WARNING ⚠ 'source' is missing. Using 'source=/usr/local/lib/python3.10/dist-packages/ultralytics/assets'.

image 1/2 /usr/local/lib/python3.10/dist-packages/ultralytics/assets/bus.jpg: 640x480 4 persons, 1 bus, 276.0ms
image 2/2 /usr/local/lib/python3.10/dist-packages/ultralytics/assets/zidane.jpg: 384x640 2 persons, 1 tie, 140.2ms
Speed: 3.8ms preprocess, 208.1ms inference, 3.9ms postprocess per image at shape (1, 3, 384, 640)
Results saved to runs/detect/train2
2 labels saved to runs/detect/train2/labels
[ultralytics.engine.results.Results object with attributes:

 boxes: ultralytics.engine.results.Boxes object
 keypoints: None
 masks: None
 names: {0: 'person', 1: 'bicycle', 2: 'car', 3: 'motorcycle', 4: 'airplane', 5: 'bus', 6: 'train', 7: 'truck', 8: 'boat', 9: 'traffic light', 10:
'fire hydrant', 11: 'stop sign', 12: 'parking meter', 13: 'bench', 14: 'bird', 15: 'cat', 16: 'dog', 17: 'horse', 18: 'sheep', 19: 'cow', 20:
'elephant', 21: 'bear', 22: 'zebra', 23: 'giraffe', 24: 'backpack', 25: 'umbrella', 26: 'handbag', 27: 'tie', 28: 'suitcase', 29: 'frisbee', 30:
'skis', 31: 'snowboard', 32: 'sports ball', 33: 'kite', 34: 'baseball bat', 35: 'baseball glove', 36: 'skateboard', 37: 'surfboard', 38: 'tennis
racket', 39: 'bottle', 40: 'wine glass', 41: 'cup', 42: 'fork', 43: 'knife', 44: 'spoon', 45: 'bowl', 46: 'banana', 47: 'apple', 48: 'sandwich',
49: 'orange', 50: 'broccoli', 51: 'carrot', 52: 'hot dog', 53: 'pizza', 54: 'donut', 55: 'cake', 56: 'chair', 57: 'couch', 58: 'potted plant', 59:
'bed', 60: 'dining table', 61: 'toilet', 62: 'tv', 63: 'laptop', 64: 'mouse', 65: 'remote', 66: 'keyboard', 67: 'cell phone', 68: 'microwave', 69:
'oven', 70: 'toaster', 71: 'sink', 72: 'refrigerator', 73: 'book', 74: 'clock', 75: 'vase', 76: 'scissors', 77: 'teddy bear', 78: 'hair drier',
79: 'toothbrush'}
 obb: None
 orig_img: array([[[119, 146, 172],
         [121, 148, 174],
         [122, 152, 177],
         ...,
         [161, 171, 188],
         [160, 170, 187],
         [160, 170, 187]],

        [[120, 147, 173],
         [122, 149, 175],
         [123, 153, 178],
         ...,
         [161, 171, 188],
         [160, 170, 187],
         [160, 170, 187]],

        [[123, 150, 176],
         [124, 151, 177],
         [125, 155, 180],
         ...,
         [161, 171, 188],
         [160, 170, 187],
         [160, 170, 187]],

        ...,

        [[183, 182, 186],
         [179, 178, 182],
         [180, 179, 183],
         ...,
         [121, 111, 117],
         [113, 103, 109],
         [115, 105, 111]],

        [[165, 164, 168],
         [173, 172, 176],
```

코딩을 시작하거나 AI로 코드를 생성하세요.