輸入執行指令

接下來是輸入各項指令，千萬注意輸入內容和輸入環境

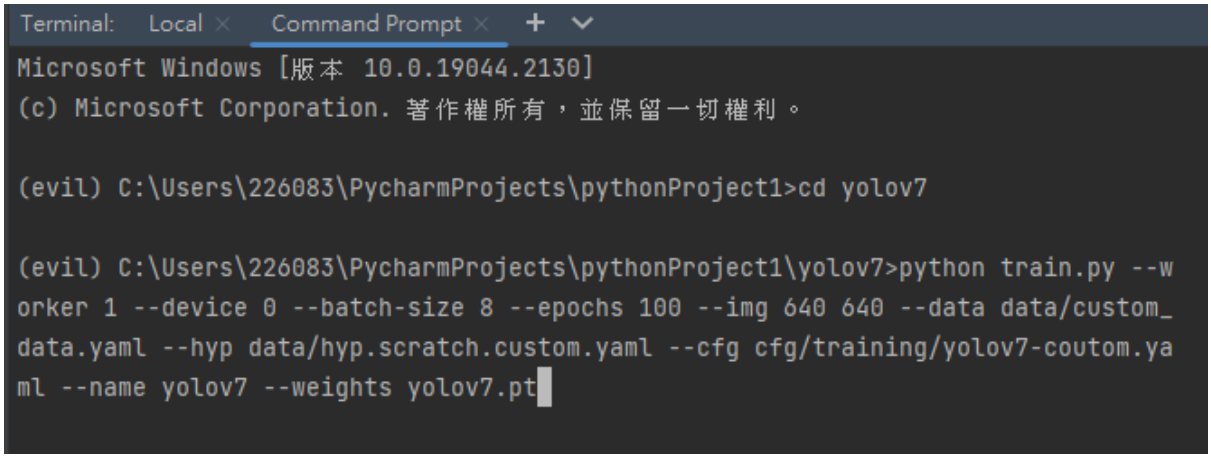
內容錯誤例如：

worker 1，中間有空格

batch-siza，明顯錯誤，改成size

環境錯誤：

套件都在yolov7這個資料夾環境下，所以當然是在這邊執行



指令如下：

python train.py --worker 1 --device 0 --batch-size 8 --epochs 100 --img 640 640 --data data/custom_data.yaml --hyp data/hyp.scratch.custom.yaml --cfg cfg/training/yolov7-custom.yaml --name yolov7 --weights yolov7.pt

python train.py：訓練過程執行train.py這個檔案，不是detect.py

data data/custom_data.yaml：決定訓練圖片的路徑和標籤數目、種類

hyp data/hyp.scratch.custom.yaml：各項參數，直接用就好，暫時別去改動

cfg cfg/training/yolov7-coutom.yaml：演算法相關，前面針對這檔案作出標籤數量的改動(80→2)

name yolov7：你的環境資料夾名稱

weights yolov7.pt：使用的模型種類，選用yolov7.pt，之後可以考慮換成其他模型

hyp.scratch.custom.yaml

```yaml
lr0: 0.01  # initial learning rate (SGD=1E-2, Adam=1E-3)
lrf: 0.1  # final OneCycleLR learning rate (lr0 * lrf)
momentum: 0.937  # SGD momentum/Adam beta1
weight_decay: 0.0005  # optimizer weight decay 5e-4
warmup_epochs: 3.0  # warmup epochs (fractions ok)
warmup_momentum: 0.8  # warmup initial momentum
warmup_bias_lr: 0.1  # warmup initial bias lr
box: 0.05  # box loss gain
cls: 0.3  # cls loss gain
cls_pw: 1.0  # cls BCELoss positive_weight
obj: 0.7  # obj loss gain (scale with pixels)
obj_pw: 1.0  # obj BCELoss positive_weight
iou_t: 0.20  # IoU training threshold
anchor_t: 4.0  # anchor-multiple threshold
# anchors: 3  # anchors per output layer (0 to ignore)
fl_gamma: 0.0  # focal loss gamma (efficientDet default gamma=1.5)
hsv_h: 0.015  # image HSV-Hue augmentation (fraction)
hsv_s: 0.7  # image HSV-Saturation augmentation (fraction)
hsv_v: 0.4  # image HSV-Value augmentation (fraction)
degrees: 0.0  # image rotation (+/- deg)
translate: 0.2  # image translation (+/- fraction)
scale: 0.5  # image scale (+/- gain)
shear: 0.0  # image shear (+/- deg)
perspective: 0.0  # image perspective (+/- fraction), range 0-0.001
flipud: 0.0  # image flip up-down (probability)
fliplr: 0.5  # image flip left-right (probability)
mosaic: 1.0  # image mosaic (probability)
mixup: 0.0  # image mixup (probability)
copy_paste: 0.0  # image copy paste (probability)
paste_in: 0.0  # image copy paste (probability), use 0 for faster training
loss_ota: 1 # use ComputeLossOTA, use 0 for faster training
```

==========

確認輸入內容正確後，開始執行，等她跑完