

Got it, Mokshitha! Since YOLOv7 is **already downloaded on your Linux system**, and you're working **offline**, I'll now give you a **very clear step-by-step plan with proper Linux commands** and **simple explanations** — from start to training your CAT model using YOLOv7.

FULL WORKFLOW: YOLOv7 CAT MODEL TRAINING (Linux Offline)

◆ ASSUMPTIONS:

- YOLOv7 folder is already there (e.g., /home/mokshitha/yolov7)
 - You're training on annotated CAT images
 - Internet is **not available**
-

STEP 1: Go to the YOLOv7 Folder

This is where all scripts are already available.

```
cd /home/mokshitha/yolov7
```

- Use `pwd` to verify you're inside `yolov7`.
-

STEP 2: Create Dataset Folders

You'll need folders for images and labels.

```
mkdir -p images/train images/val
```

```
mkdir -p labels/train labels/val
```

- ◆ `-p` creates the full path in one command.
-

STEP 3: Put Your Dataset Files

Copy your CAT image files (e.g., .jpg or .png) and label .txt files to these folders:

Folder	What goes inside
--------	------------------

images/train/	Training images
---------------	-----------------

images/val/	Validation images
-------------	-------------------

labels/train/	.txt label files (same names as images)
---------------	---

Folder	What goes inside
labels/val/	.txt label files (for val images)

 The .txt files must be in **YOLO format**:

<class_id> <x_center> <y_center> <width> <height>

All values are normalized between 0 and 1.

STEP 4: Create YAML File (cat.yaml)

Create this file inside the data/ folder.

```
mkdir -p data
```

```
nano data/cat.yaml
```

Paste this inside cat.yaml:

```
train: /home/mokshitha/yolov7/images/train
```

```
val: /home/mokshitha/yolov7/images/val
```

```
nc: 1
```

```
names: ['cat']
```

 Save and exit in nano: Press CTRL + O, then ENTER, then CTRL + X

STEP 5: Check for YOLOv7 Weights

If yolov7.pt file already exists in your folder, you're good.

Check:

```
ls yolov7.pt
```

If it shows the file, you're ready.

STEP 6: Train the Model

Now run the training command:

```
python3 train.py --img 640 --batch 16 --epochs 100 --data data/cat.yaml --cfg  
cfg/training/yolov7.yaml --weights yolov7.pt --name yolov7_cat
```

- 🟡 What each part means:

Flag	Meaning
--img 640	Image size
--batch 16	Batch size
--epochs 100	Number of training cycles
--data	Path to your cat.yaml
--cfg	Network architecture
--weights	Pretrained weights
--name	Name of output folder (in runs/train/)

STEP 7: After Training (Check Results)

YOLOv7 stores your training results in:

runs/train/yolov7_cat/

Use this to check logs and your trained model (best.pt).

STEP 8: Run Inference on New Images

Once training is done, test it like this:

```
python3 detect.py --weights runs/train/yolov7_cat/weights/best.pt --conf 0.25 --img 640 --source /home/mokshitha/test_images/
```

- 🟡 Output images with bounding boxes will be saved in:

runs/detect/exp/

ALL COMMANDS TO COPY & RUN

```
# Go to YOLOv7 folder  
cd /home/mokshitha/yolov7
```

```
# Create dataset folders
```

```
mkdir -p images/train images/val  
mkdir -p labels/train labels/val  
  
# Make data folder and cat.yaml  
mkdir -p data  
nano data/cat.yaml  
  
# (Paste YAML contents, save and exit)  
  
# Confirm weights are available  
ls yolov7.pt  
  
# Train model  
python3 train.py --img 640 --batch 16 --epochs 100 --data data/cat.yaml --cfg  
cfg/training/yolov7.yaml --weights yolov7.pt --name yolov7_cat  
  
# Run inference  
python3 detect.py --weights runs/train/yolov7_cat/weights/best.pt --conf 0.25 --img 640 --  
source /home/mokshitha/test_images/
```

You're Now Training YOLOv7 on Linux Offline Successfully

Let me know if:

- You want me to generate a **PDF version** of this new guide
- You want help checking if your labels are in YOLO format

I'm right here to walk you through.