

## ✓ TASK1 Rename notebook

For GNU/Linux box:

1. Create virtual environment <https://docs.python.org/3/library/venv.html>,
2. \$ source [/path/to/venv/bin/activate](#)
3. Ensure you are inside your virtual environment.. before proceeding
4. Install ultralytics - follow steps given at <https://docs.ultralytics.com/quickstart/>
5. Test installation with the example given, and verify whether the trained model can recognise objects and also perform segmentation from the online image.

## ✓ Procedure

### ✓ Step 1

I created a virtual environment in my current working Directory through:

```
python -m venv venv
```

### ✓ Step 2

Activated the Virtual Environment through:

```
.\venv\Scripts\activate
```

### ✓ Step 3

Installed the Ultralytics with environment activated through:

```
pip install ultralytics
```

### ✓ Step 4

With below python script predicted the objects in images using YOLOv8

```
from ultralytics import YOLO as Y

# Load models
```

```
m = Y('yolov8n.pt')
m1 = Y('yolov8n-seg.pt')
m2 = Y('yolov8n.pt')
m3 = Y('yolov8m.pt')
m4 = Y('yolov8x.pt')
m5 = Y('yolov8l.pt')

# Predict and save results to custom folders
m1.predict(source="https://ultralytics.com/images/bus.jpg", save=True, name="nano-seg")
m.predict(source="bus.jpg", save=True, name="nano")
m2.predict(source="bus.jpg", save=True, name="small")
m3.predict(source="bus.jpg", save=True, name="medium")
m4.predict(source="bus.jpg", save=True, name="xlarge")
m5.predict(source="bus.jpg", save=True, name="large")
#predicting the test image
m1.predict(source="testing.jpg", save=True, name="nano-seg_test")
m.predict(source="testing.jpg", save=True, name="nano_test")
m2.predict(source="testing.jpg", save=True, name="small_test")
m3.predict(source="testing.jpg", save=True, name="medium_test")
m4.predict(source="testing.jpg", save=True, name="xlarge_test")
m5.predict(source="testing.jpg", save=True, name="large_test")
```

## ▼ Step 5

Above code is used to predict objects in two images and the output differed based on the model used (nano,small,medium,large,extra\_large).Below two images are the output of the tests:





