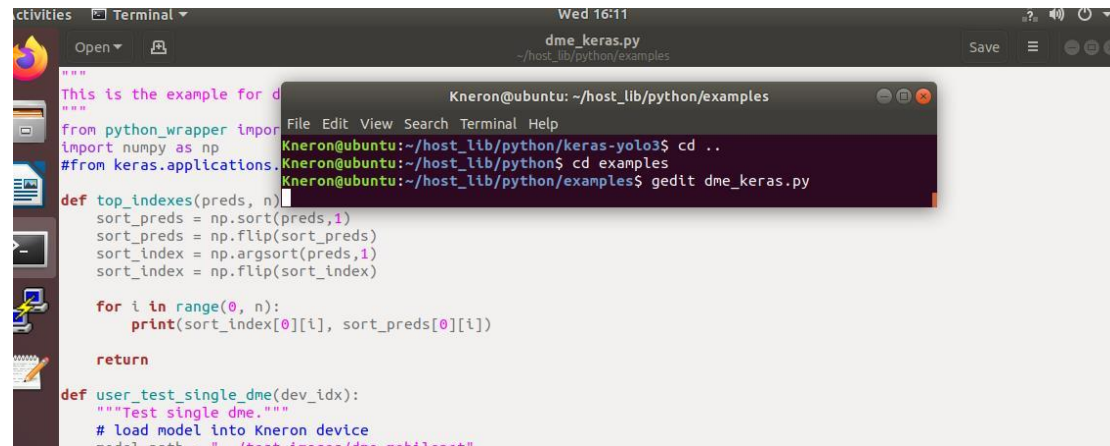


Go to /host\_lib/python/examples

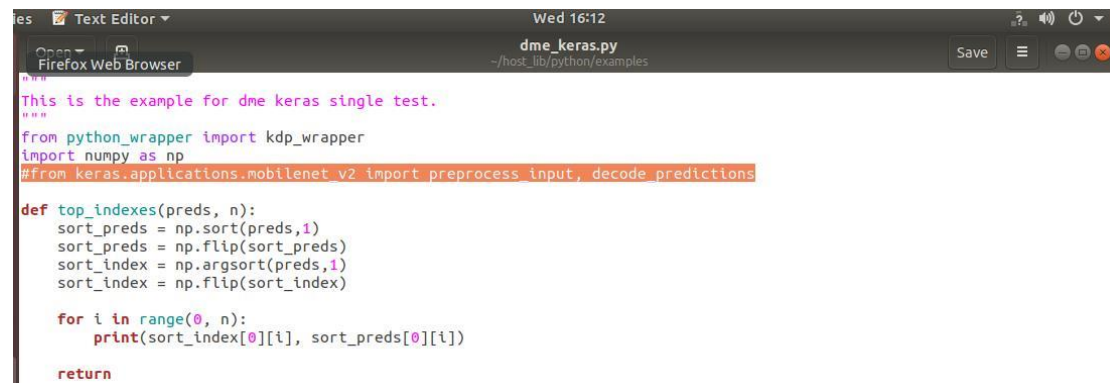
Use gedit dme\_keras.py to open dme\_keras.py



```
Wed 16:11
dme_keras.py
~/host_lib/python/examples

Kneron@ubuntu: ~/host_lib/python/examples
File Edit View Search Terminal Help
Kneron@ubuntu:~/host_lib/python/keras-yolo3$ cd ..
Kneron@ubuntu:~/host_lib/python$ cd examples
Kneron@ubuntu:~/host_lib/python/examples$ gedit dme_keras.py
```

Remove the “#” from marked line



```
Wed 16:12
dme_keras.py
~/host_lib/python/examples

This is the example for dme keras single test.
"""
from python_wrapper import kdp_wrapper
import numpy as np
#from keras.applications.mobilenet_v2 import preprocess_input, decode_predictions

def top_indexes(preds, n):
    sort_preds = np.sort(preds,1)
    sort_preds = np.flip(sort_preds)
    sort_index = np.argsort(preds,1)
    sort_index = np.flip(sort_index)

    for i in range(0, n):
        print(sort_index[0][i], sort_preds[0][i])

    return
```

Text Editor Wed 16:27

Open  \*dme\_keras.py  
~/host\_lib/python/examples

```

"""
This is the example for dme keras single test.
"""
from python_wrapper import kdp_wrapper
import numpy as np
from keras.applications.mobilenet_v2 import preprocess_input, decode_predictions

def top_indexes(preds, n):
    sort_preds = np.sort(preds,1)
    sort_preds = np.flip(sort_preds)
    sort_index = np.argsort(preds,1)
    sort_index = np.flip(sort_index)

    for i in range(0, n):
        print(sort_index[0][i], sort_preds[0][i])

    return

def user_test_single_dme(dev_idx):
    """Test single dme."""
    # load model into Kneron device
    model_path = "../test_images/dme_mobilenet"
    kdp_wrapper.kdp_dme_load_model(dev_idx, model_path)

    #get test images ready
    img_path = './data/images/index.jpeg'
    img_path2 = './data/images/car.jpeg'

    npraw_data = kdp_wrapper.kdp_inference(dev_idx, img_path)

    # Do postprocessing with keras
    preds = kdp_wrapper.softmax(npraw_data[0]).reshape(1, 1000)
    top_indexes(preds, 3)
    #print('\nPredicted:', decode_predictions(preds, top=3)[0])


    npraw_data = kdp_wrapper.kdp_inference(dev_idx, img_path2)

    # Do postprocessing with keras
    preds = kdp_wrapper.softmax(npraw_data[0]).reshape(1, 1000)
    top_indexes(preds, 3)
    #print('\nPredicted:', decode_predictions(preds, top=3)[0])

    kdp_wrapper.kdp_exit_dme(dev_idx)

```

s Text Editor Wed 16:28

Open  \*dme\_keras.py  
~/host\_lib/python/examples

```

"""
This is the example for dme keras single test.
"""
from python_wrapper import kdp_wrapper
import numpy as np
from keras.applications.mobilenet_v2 import preprocess_input, decode_predictions

def top_indexes(preds, n):
    sort_preds = np.sort(preds,1)
    sort_preds = np.flip(sort_preds)
    sort_index = np.argsort(preds,1)
    sort_index = np.flip(sort_index)

    for i in range(0, n):
        print(sort_index[0][i], sort_preds[0][i])

    return

def user_test_single_dme(dev_idx):
    """
    single dme.
    """
    # Load model into Kneron device
    model_path = "../test_images/dme_mobilenet"
    kdp_wrapper.kdp_dme_load_model(dev_idx, model_path)

    #get test images ready
    img_path = './data/images/index.jpeg'
    img_path2 = './data/images/car.jpeg'

    npraw_data = kdp_wrapper.kdp_inference(dev_idx, img_path)

    # Do postprocessing with keras
    preds = kdp_wrapper.softmax(npraw_data[0]).reshape(1, 1000)
    top_indexes(preds, 3)
    #print('\nPredicted:', decode_predictions(preds, top=3)[0])

    npraw_data = kdp_wrapper.kdp_inference(dev_idx, img_path2)

    # Do postprocessing with keras
    preds = kdp_wrapper.softmax(npraw_data[0]).reshape(1, 1000)
    top_indexes(preds, 3)
    #print('\nPredicted:', decode_predictions(preds, top=3)[0])

```

```

s Text Editor Wed 16:29
Open Firefox Web Browser *dme_keras.py
~/host_lib/python/examples

"""
This is the example for dme keras single test.
"""
from python_wrapper import kdp_wrapper
import numpy as np
from keras.applications.mobilenet_v2 import preprocess_input, decode_predictions

def top_indexes(preds, n):
    sort_preds = np.sort(preds,1)
    sort_preds = np.flip(sort_preds)
    sort_index = np.argsort(preds,1)
    sort_index = np.flip(sort_index)

    for i in range(0, n):
        print(sort_index[0][i], sort_preds[0][i])

    return

def user_test_single_dme(dev_idx):
    """Test single dme."""
    # load model into Kneron device
    model_path = "../test_images/dme_mobilenet"
    kdp_wrapper.kdp_dme_load_model(dev_idx, model_path)

    #get test images ready
    img_path = './data/images/index.jpeg'
    img_path2 = './data/images/car.jpeg'

    nprw_data = kdp_wrapper.kdp_inference(dev_idx, img_path)

    # Do postprocessing with keras
    preds = kdp_wrapper.softmax(nprw_data[0]).reshape(1, 1000)
    top_indexes(preds, 3)
    print('\nPredicted:', decode_predictions(preds, top=3)[0])

    nprw_data = kdp_wrapper.kdp_inference(dev_idx, img_path2)

    # Do postprocessing with keras
    preds = kdp_wrapper.softmax(nprw_data[0]).reshape(1, 1000)
    top_indexes(preds, 3)
    print('\nPredicted:', decode_predictions(preds, top=3)[0])

    kdp_wrapper.kdp_exit_dme(dev_idx)

```

Download two images into /host\_lib/python/data/images , and change the marked line to downloaded images' name

```
es Text Editor Wed 16:18
*dme_keras.py
~/host_lib/python/examples
Save

This is the example for dme keras single test.
"""
from python_wrapper import kdp_wrapper
import numpy as np
from keras.applications.mobilenet_v2 import preprocess_input, decode_predictions
print('\nPrediction : ', decode_predictions(preds, top=3)[0])

def top_indexes(preds, n):
    sort_preds = np.sort(preds,1)
    sort_preds = np.flip(sort_preds)
    sort_index = np.argsort(preds,1)
    sort_index = np.flip(sort_index)

    for i in range(0, n):
        print(sort_index[0][i], sort_preds[0][i])

    return

def user_test_single_dme(dev_idx):
    """Test single dme."""
    # load model into Kneron device
    model_path = "../test_images/dme_mobilenet"
    kdp_wrapper.kdp_dme_load_model(dev_idx, model_path)

    #get test images ready
    img_path = '../data/images/cat.jpg'
    img_path2 = '../data/images/fox.jpg'

    new_data = kdp_wrapper.kdp_inference(dev_idx, img_path)
```

```
ies Text Editor Wed 16:18
*dme_keras.py
~/host_lib/python/examples
Save

This is the example for dme keras single test.
"""
from python_wrapper import kdp_wrapper
import numpy as np
from keras.applications.mobilenet_v2 import preprocess_input, decode_predictions
print('\nPrediction : ', decode_predictions(preds, top=3)[0])

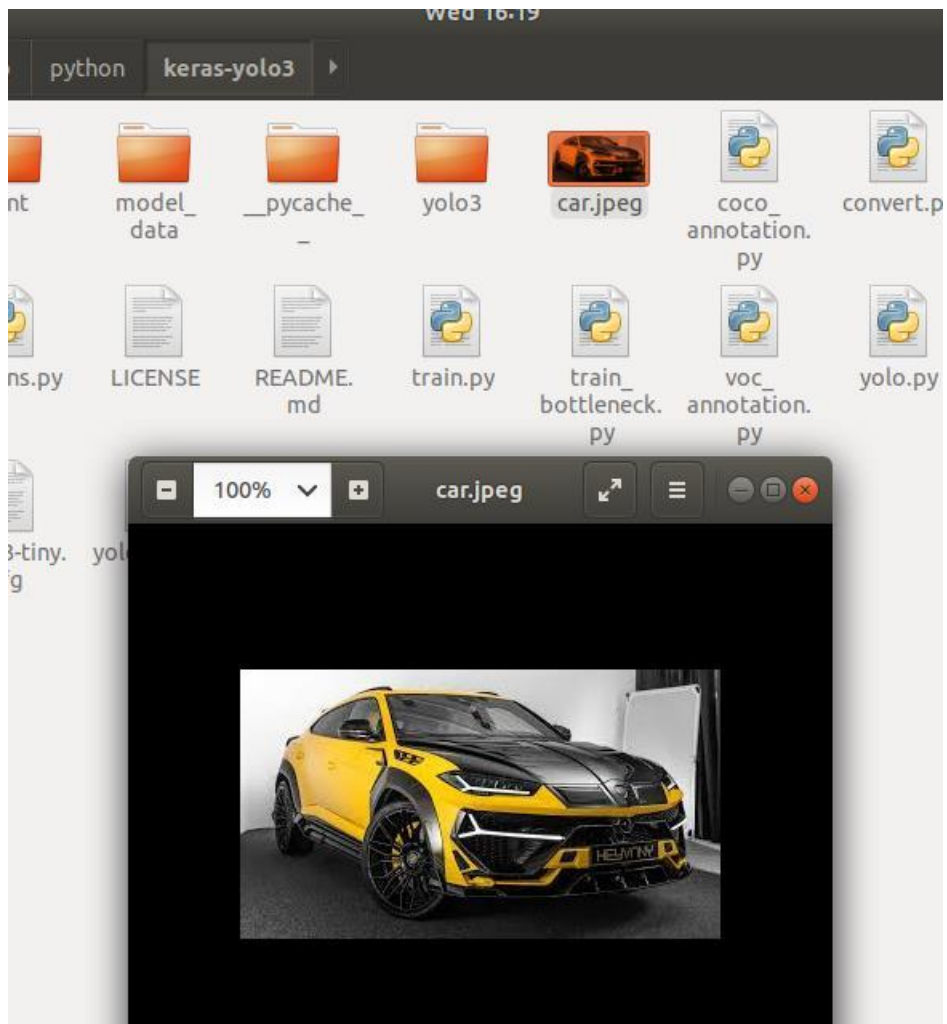
def top_indexes(preds, n):
    sort_preds = np.sort(preds,1)
    sort_preds = np.flip(sort_preds)
    sort_index = np.argsort(preds,1)
    sort_index = np.flip(sort_index)

    for i in range(0, n):
        print(sort_index[0][i], sort_preds[0][i])

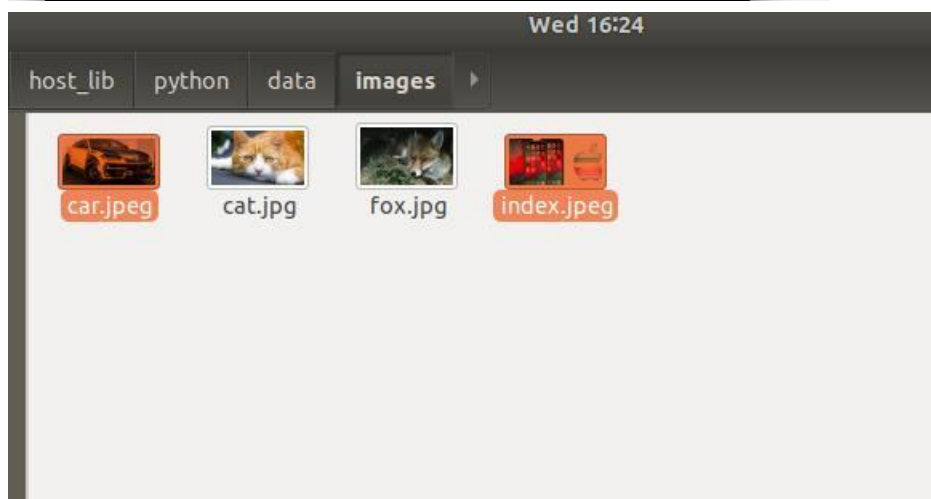
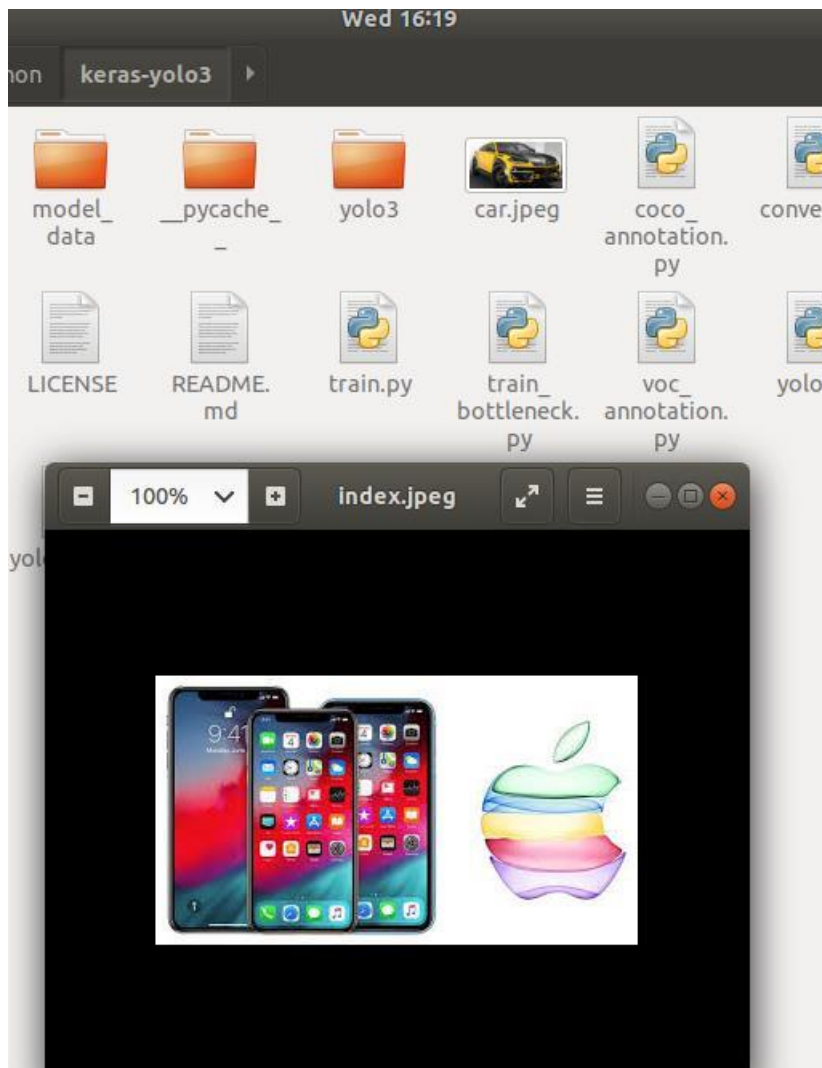
    return

def user_test_single_dme(dev_idx):
    """Test single dme."""
    # load model into Kneron device
    model_path = "../test_images/dme_mobilenet"
    kdp_wrapper.kdp_dme_load_model(dev_idx, model_path)

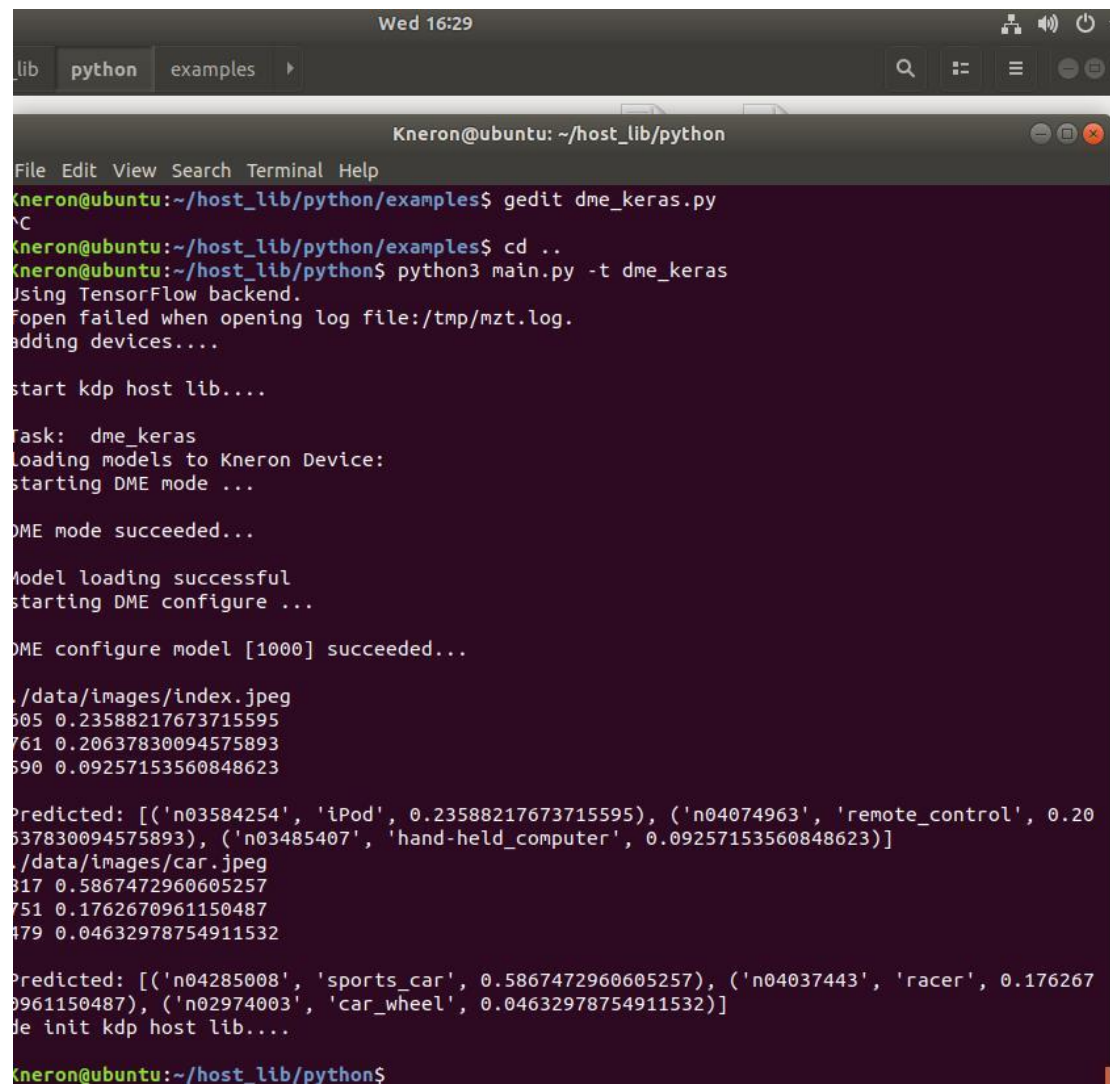
    #get test images ready
    img_path = '../data/images/index.jpeg'
    img_path2 = '../data/images/car.jpeg'
```







Go back to /host\_lib/python and enter `python3 main.py -t dme_keras`



The image shows a terminal window titled "Kneron@ubuntu: ~/host\_lib/python". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content shows the following sequence of commands and output:

```
Kneron@ubuntu:~/host_lib/python/examples$ gedit dme_keras.py
Kneron@ubuntu:~/host_lib/python/examples$ cd ..
Kneron@ubuntu:~/host_lib/python$ python3 main.py -t dme_keras
Using TensorFlow backend.
fopen failed when opening log file:/tmp/mzt.log.
Adding devices....

start kdp host lib....

Task: dme_keras
Loading models to Kneron Device:
starting DME mode ...

DME mode succeeded...

Model loading successful
starting DME configure ...

DME configure model [1000] succeeded...

./data/images/index.jpeg
505 0.23588217673715595
761 0.20637830094575893
590 0.09257153560848623

Predicted: [('n03584254', 'iPod', 0.23588217673715595), ('n04074963', 'remote_control', 0.20637830094575893), ('n03485407', 'hand-held_computer', 0.09257153560848623)]

./data/images/car.jpeg
817 0.5867472960605257
751 0.1762670961150487
479 0.04632978754911532

Predicted: [('n04285008', 'sports_car', 0.5867472960605257), ('n04037443', 'racer', 0.1762670961150487), ('n02974003', 'car_wheel', 0.04632978754911532)]
de init kdp host lib....

Kneron@ubuntu:~/host_lib/python$
```