

```

In [5]: import numpy as np
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.resnet50 import ResNet50, preprocess_input, decode_predictions

# Load the ResNet50 model
model = ResNet50(weights='imagenet')

# Define a function to predict the class of an image
def predict_image(image_path):
    img = image.load_img(image_path, target_size=(224, 224))
    x = image.img_to_array(img)
    x = np.expand_dims(x, axis=0)
    x = preprocess_input(x)
    preds = model.predict(x)
    return decode_predictions(preds, top=1)[0][0]

# Predict the class of each image and save the results
images = ['C:/Users/chris/OneDrive/Desktop/Boots.jpg', 'C:/Users/chris/OneDrive/Desktop/Slipper.jpg', 'C:/Users/chris/OneDrive/Desktop/Adidas.jpg', 'C:/Users/chris/OneDrive/Desktop/Nike.jpg', 'C:/Users/chris/OneDrive/Desktop/Ankleboots.jpg']
results = []
for image_path in images:
    result = predict_image(image_path)
    results.append(result)
    print(image_path + ': ' + result[1])
np.save('dsc650/assignments/assignment06/results/Assignment 6.3', results)

1/1 [=====] - 22s 22s/step
C:/Users/chris/OneDrive/Desktop/Boots.jpg: cowboy_boot
1/1 [=====] - 10s 10s/step
C:/Users/chris/OneDrive/Desktop/Slipper.jpg: knee_pad
1/1 [=====] - 12s 12s/step
C:/Users/chris/OneDrive/Desktop/Adidas.jpg: running_shoe
1/1 [=====] - 12s 12s/step
C:/Users/chris/OneDrive/Desktop/Nike.jpg: running_shoe
1/1 [=====] - 13s 13s/step
C:/Users/chris/OneDrive/Desktop/Ankleboots.jpg: Loafer

```

In [ ]: