In [3]:
```python
# Import necessary libraries
from tensorflow.keras.applications.resnet50 import ResNet50
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.resnet50 import preprocess_input, decode_predictions
import numpy as np
import os

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

# Directory containing images
image_dir = './images63/'

# Directory to save predictions
save_dir = './results/predictions/resnet50/'

# Create save_dir if it doesn't exist
if not os.path.exists(save_dir):
    os.makedirs(save_dir)

# Get image files
image_files = [f for f in os.listdir(image_dir) if os.path.isfile(os.path.join(image_di

# Iterate through image files and make predictions
for image_file in image_files:
    img_path = os.path.join(image_dir, image_file)
    img = image.load_img(img_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)
    # decode the results into a list of tuples (class, description, probability)
    # (one such list for each sample in the batch)
    preds_decoded = decode_predictions(preds, top=3)[0]

    # Write the predictions to a file
    with open(os.path.join(save_dir, f'{image_file}_prediction.txt'), 'w') as f:
        for i, pred in enumerate(preds_decoded):
            f.write(f"{i+1}. {pred[1]}: {pred[2]*100:.2f}%\n")
```

Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/resne
t/resnet50_weights_tf_dim_ordering_tf_kernels.h5
102973440/102967424 [==============================] - 12s 0us/step
Downloading data from https://storage.googleapis.com/download.tensorflow.org/data/imagen
et_class_index.json
40960/35363 [================================] - 0s 1us/step