Assignment 6.3

In [27]:
```
import numpy as np
from pathlib import Path
import os
from keras.preprocessing.image import image
from keras.preprocessing.image import img_to_array
from keras.applications.resnet50 import preprocess_input, ResNet50, decode_pre
dictions
from keras.applications.imagenet_utils import decode_predictions
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
```

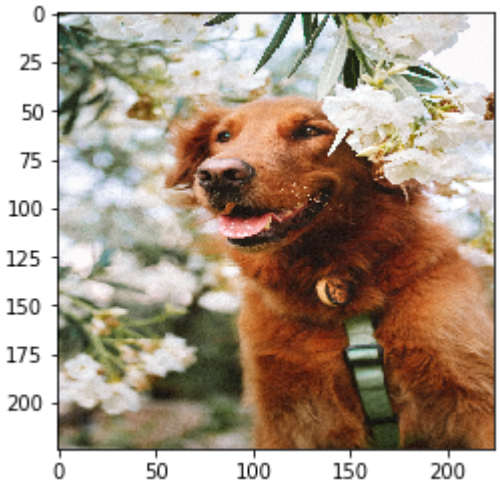
Load Data Set

In [28]:
```
current_dir = Path(os.getcwd()).absolute()
images_dir = current_dir.joinpath('images')
```

In [29]:
```
for root, dirs, files in os.walk(images_dir):
    for file_path in files:
        ## Current path is now the file path to the image.
        current_path = Path(root).joinpath(file_path)
        break

img = image.load_img(current_path, target_size=(224, 224))
plt.imshow(img)
```

Out[29]: <matplotlib.image.AxesImage at 0x22c296f9b70>

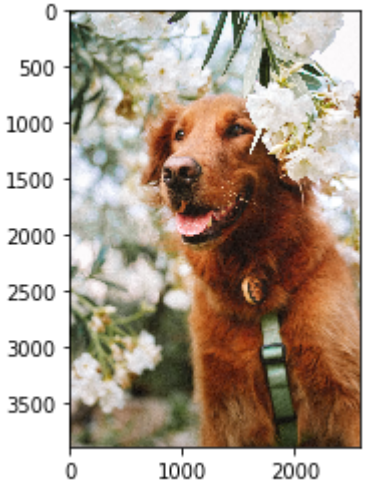


In [36]:
```
def image_processing(img_path):
    img = image.load_img(img_path, target_size = (224, 224))
    # image to array
    img = image.img_to_array(img)
    img = np.expand_dims(img, axis=0)
    img = preprocess_input(img)
    return img
```

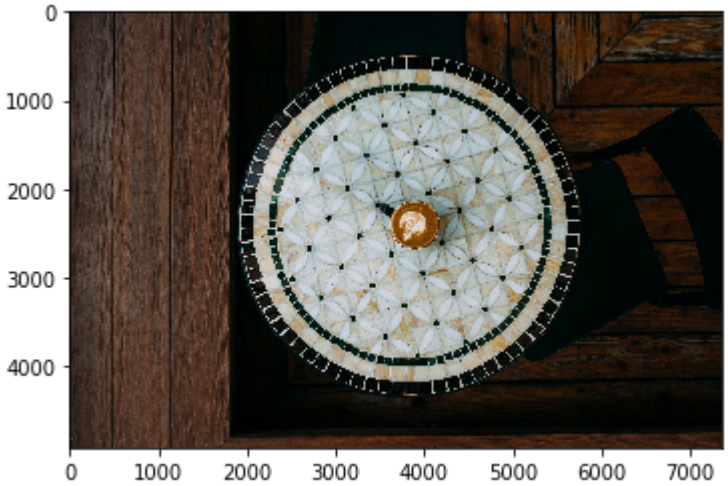
In [37]:
```
def image_classfier(processed_img):
    preds = model.predict(processed_img)
    prediction = decode_predictions(preds, top=1)[0][0]
    return prediction
```

In [38]:
```
model = ResNet50(weights='imagenet')
```

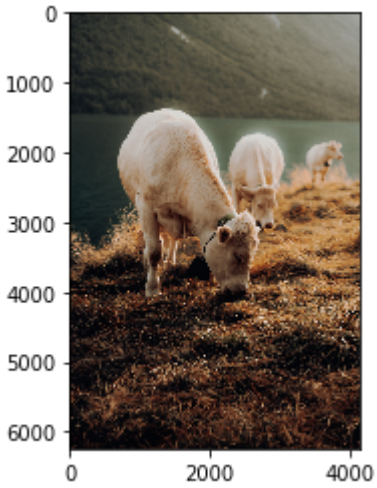
In [41]:
```
for root, dirs, pictures in os.walk(images_dir):
    for picture in pictures:
        image_path = Path(root).joinpath(picture)
        img = image_processing(image_path)
        prediction = image_classfier(img)
        pic = mpimg.imread(image_path)
        plt.imshow(pic)
        plt.show()
        print(prediction)
```



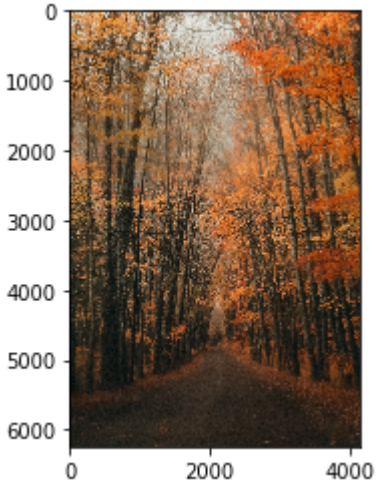
('n02093991', 'Irish_terrier', 0.2301139)



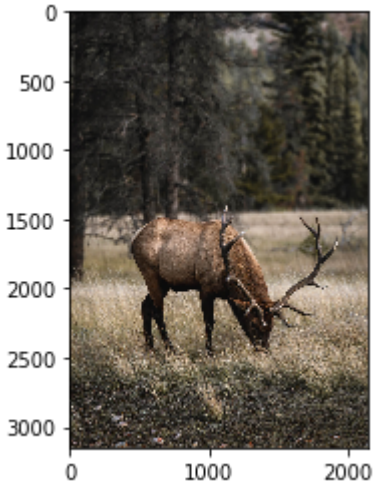
('n04328186', 'stopwatch', 0.40761834)



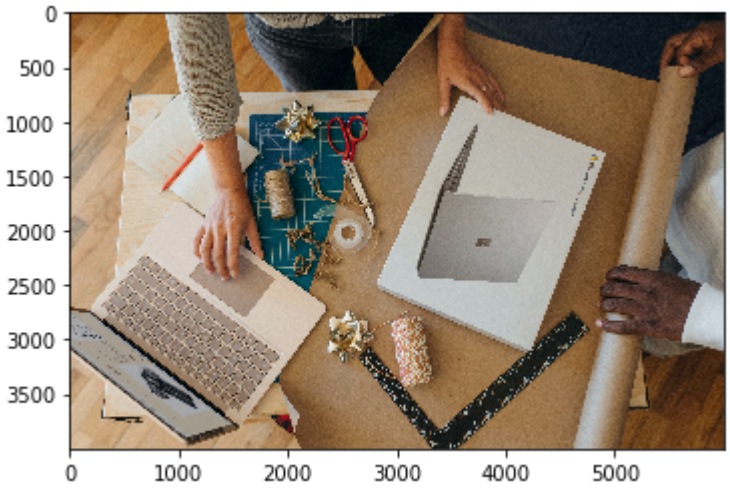
('n02395406', 'hog', 0.75355124)



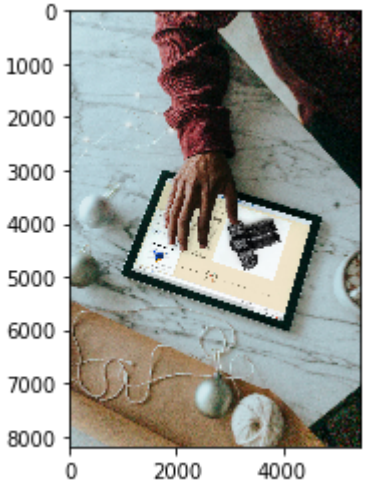

```
('n02793495', 'barn', 0.23763092)
```




```
('n02410509', 'bison', 0.38129005)
```



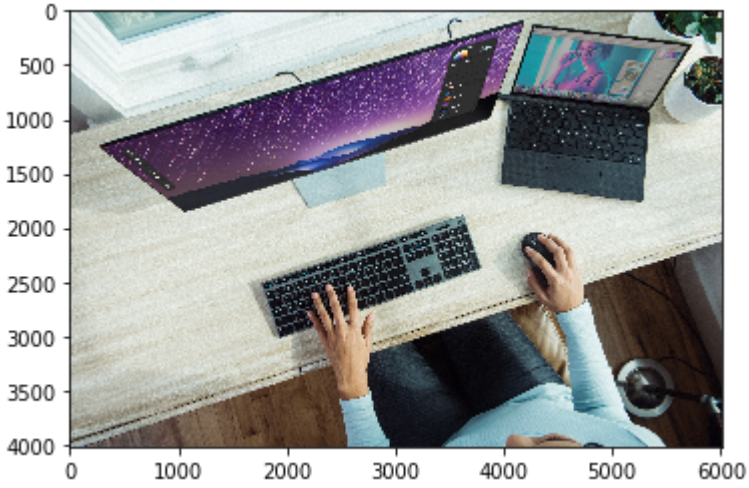

```
('n03000247', 'chain_mail', 0.2248758)
```




```
('n03223299', 'doormat', 0.4948146)
```


```
('n04590129', 'window_shade', 0.2625627)
```


```
('n03899768', 'patio', 0.89413834)
```




```
('n04548362', 'wallet', 0.65800023)
```

In [ ]: