```
from google.colab import files
import os


input_path = "/content/drive/MyDrive/Y1715.jpg"
output_path = "/content/drive/MyDrive/IMG-20250125-WA0001.jpg"

print("□ Folders created!")
print(" Now upload your 5 images.")
```

```
□ Folders created!
 Now upload your 5 images.
```

```
uploaded = files.upload()
```

```
<IPython.core.display.HTML object>
```

```
Saving Y1720.jpg to Y1720.jpg
Saving Y1717.jpg to Y1717.jpg
Saving Y1716.jpg to Y1716.jpg
Saving Y1715.jpg to Y1715.jpg
Saving Y1714.jpg to Y1714.jpg
```

```
import shutil # Needed for shutil.move, as it's defined later in the
notebook

# The variable 'input_path' is currently defined as a file path (e.g.,
"/content/drive/MyDrive/Y1715.jpg").
# However, for storing multiple uploaded images, it should ideally be
a directory path.
# We will extract the directory part from 'input_path' to use as the
destination folder.
# This ensures files are moved to a valid directory in Google Drive.
destination_folder = os.path.dirname(input_path)

# Create the destination folder if it doesn't exist
os.makedirs(destination_folder, exist_ok=True)

for filename in uploaded.keys():
    source_path = filename # files.upload() puts files directly in
/content/
    target_path = os.path.join(destination_folder, filename)

    # Use shutil.move to move the file. This works across different
file systems.
    shutil.move(source_path, target_path)

print("✔  Images uploaded to:", destination_folder)

# Note: The 'input_path' variable itself still refers to a file.
# You might want to consider redefining 'input_path' in the previous
```

```
cell
# (e.g., cell TcKZ2dv_jAJI) to be the directory path (e.g.,
'/content/drive/MyDrive/input_images/')
# if you intend to use it as a directory for subsequent operations.
```

```
✔  Images uploaded to: /content/drive/MyDrive
```

```
from tensorflow.keras.preprocessing.image import
ImageDataGenerator,img_to_array, load_img

datagen = ImageDataGenerator(
rotation_range=30,
width_shift_range=0.2,
height_shift_range=0.2,
zoom_range=0.2,
shear_range=0.2,
horizontal_flip=True
)

image_extensions = ('.png', '.jpg', '.jpeg', '.gif', '.bmp', '.tiff')

for img_name in os.listdir(destination_folder):
    if img_name.lower().endswith(image_extensions):
        img = load_img(os.path.join(destination_folder, img_name))
        x = img_to_array(img)
        x = x.reshape((1,) + x.shape)
    else:
        print(f"Skipping non-image file: {img_name}")
```

```
Skipping non-image file: gobinath m pass.pdf
Skipping non-image file: Colab Notebooks
Skipping non-image file: brain_tumor_dataset
Skipping non-image file: BRAINTUMOR.h5
Skipping non-image file: alzheimer
Skipping non-image file: guru
Skipping non-image file: gobi
Skipping non-image file: internship report.gdoc
Skipping non-image file: Pandas.pdf
Skipping non-image file: diabetcsv.csv
Skipping non-image file: Salary Data.csv
Skipping non-image file: grades_withnulls.csv
Skipping non-image file: diabetes (1).xlsx
Skipping non-image file: onlynum.txt
Skipping non-image file:
vertopal.com_augumented_dataset_for_fruits.gdoc
```

```
import os

# The current output_path is a file path. We need a directory for
augmented images.
# Let's create a new directory for augmented images in the same parent
```

```
folder.
augmented_output_dir = os.path.join(os.path.dirname(output_path),
"augmented_images_output")
os.makedirs(augmented_output_dir, exist_ok=True)

i = 0
for batch in datagen.flow(
    x,
    batch_size=1,
    save_to_dir=augmented_output_dir, # Use the new directory
    save_prefix="aug",
    save_format="jpg"
):
    i += 1
    if i >= 10:
        break
print(" Image augmentation complete!")
print(" Check your augmented images in:", augmented_output_dir)
```

```
 Image augmentation complete!
 Check your augmented images in:
/content/drive/MyDrive/augmented_images_output
```

```
import shutil

zip_path = "/content/augmented_images.zip"
shutil.make_archive("/content/augmented_images", 'zip',
augmented_output_dir)
```

```
{"type":"string"}
```

```
print(" Ready to download: augmented_images.zip")
```

```
 Ready to download: augmented_images.zip
```