

```

% Step 1: Loading the image lena
image = imread('/MATLAB Drive/lena.png');

% Step 2: Converting the image lena to grayscale
grayImage = rgb2gray(image);

% Step 3: Normalizing the grayscale values to range [0, 1]
normalizedImage = double(grayImage) / 255;

% Step 4: Using the imresize function to quantize the grayscale levels
% Resizing the image lena by a factor that maps the 256 levels to 32 levels
resizeFactor = 32 / 256;

% Using the nearest neighbor interpolation for resizing
resizedImage = imresize(normalizedImage, resizeFactor, 'nearest');

% Step 5: Resizing back to the original image size
quantizedImage = imresize(resizedImage, size(normalizedImage), 'nearest');

% Step 6: Mapping the image lena to 32 levels
quantizedImage = round(quantizedImage * 31);

% Step 7: Scaling the result back to [0, 255] and converting to uint8
finalQuantizedImage = uint8(quantizedImage * (255 / 31));

% Step 8: Displaying the original and quantized images below
figure;
subplot(1, 2, 1), imshow(grayImage), title('Original Grayscale Image');
subplot(1, 2, 2), imshow(finalQuantizedImage), title('Quantized to 32
Levels');

```

**Original Grayscale Image**



**Quantized to 32 Levels**



Below are the steps followed in process:

- >> Read the image and converted it to grayscale because my image lena is in color.
- >> Normalized the grayscale values so that they range between 0 and 1.
- >> Used the imresize function to reduce the number of grayscale levels.
- >> Scaled back to 32 levels: Multiplied the normalized image by 31 to map it to 32 levels of intensity.
- >> Converted the resulting image lena back to an 8-bit unsigned integer format for display.
- >> The last step displayed the original and quantized images.