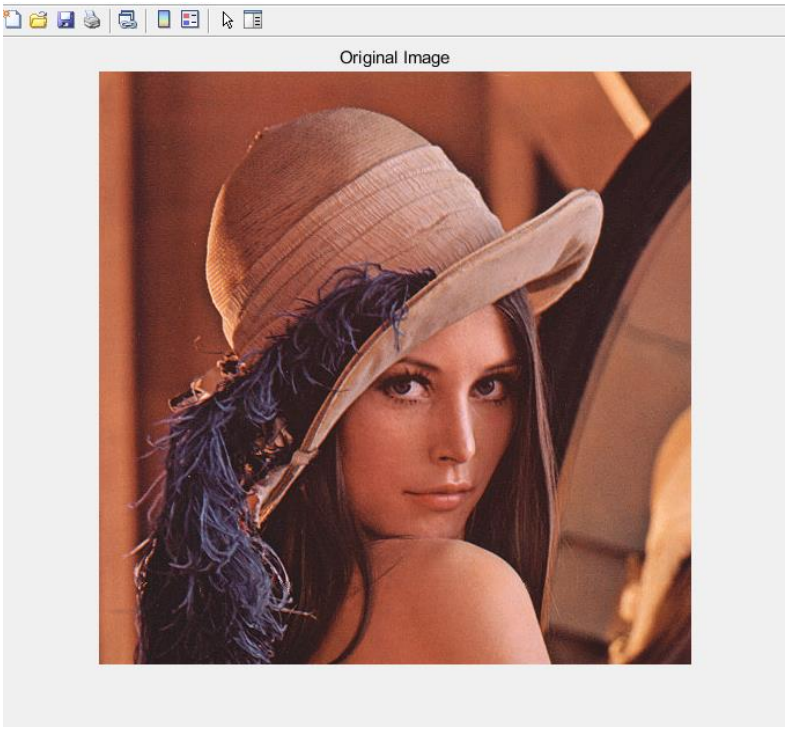
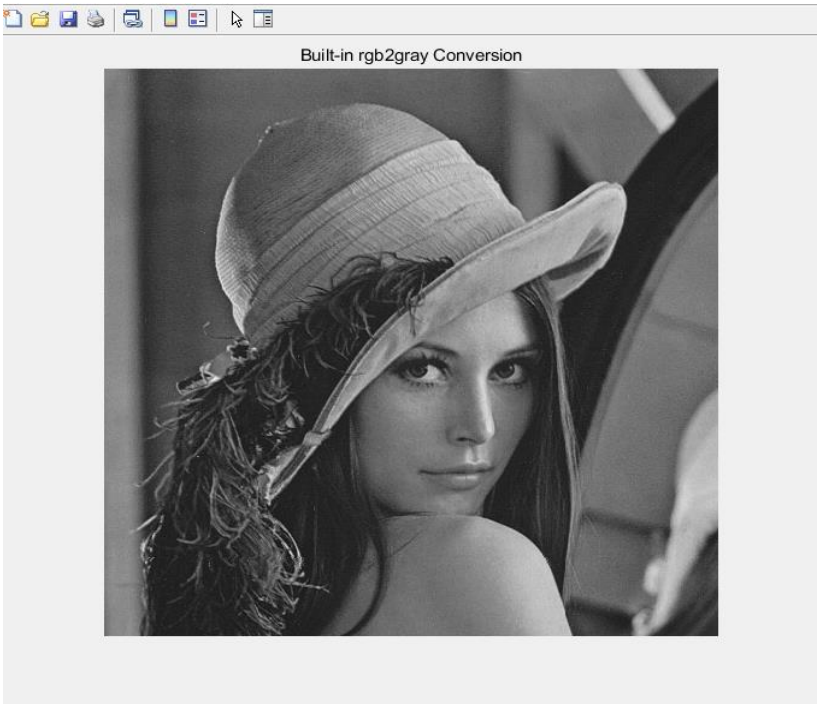
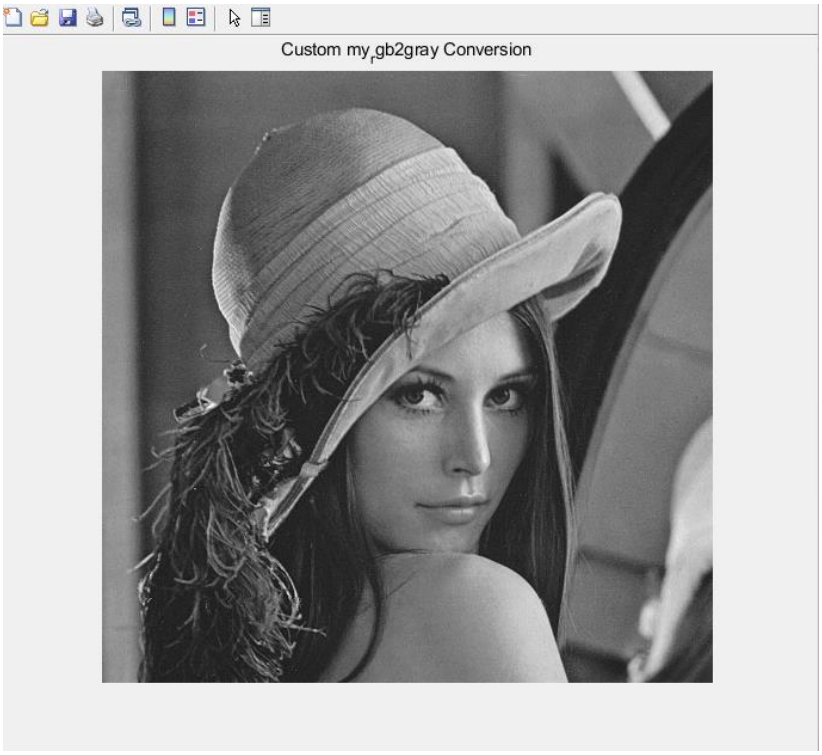
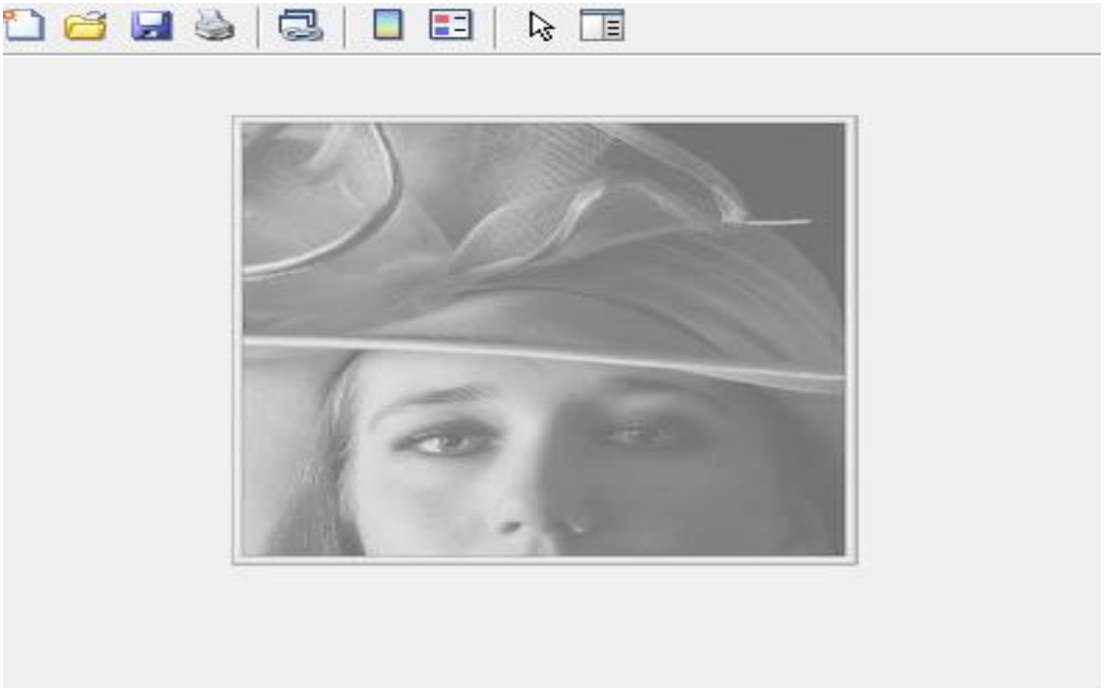
```
%% Q)1

% a) Read and show the image lena.bmp
lena = imread('lena.bmp');
figure, imshow(lena);
title('Original Image');

% b) Convert the image into gray-scale
lena_gray = rgb2gray(lena);
figure, imshow(lena_gray);
title('Built-in rgb2gray Conversion');

% c) Write my own function my_rgb2gray to convert an RGB image to
grayscale

lena_custom_gray = my_rgb2gray(lena);
figure, imshow(lena_custom_gray);
title('Custom my_rgb2gray Conversion');

% d) Save the above gray-scale image to a file named lena_gray.jpg.
imwrite(lena_custom_gray, 'lena_gray.jpg');
```

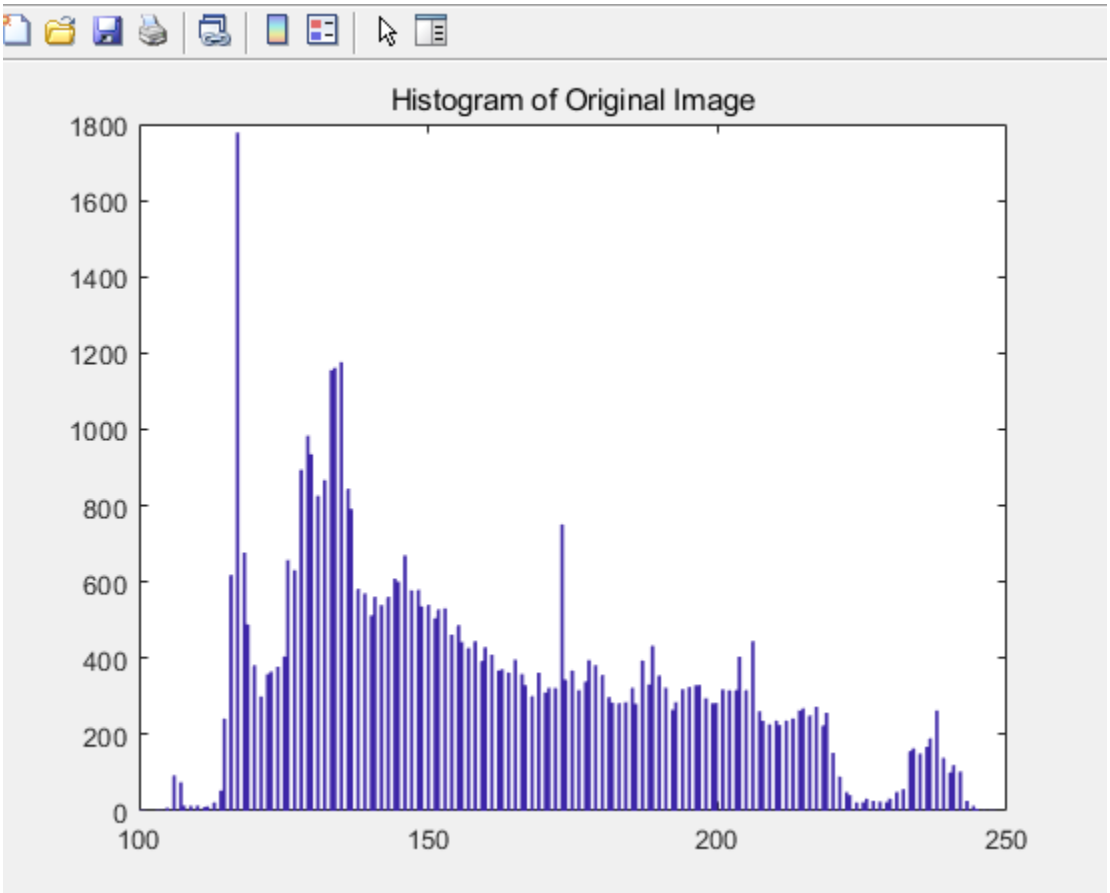



Built-in rgb2gray Conversion


Custom my_gb2gray Conversion

```
%% Q)2

% Read and show the image lowcontrast.jpg
lowcontrast = imread('lowcontrast.jpg');
imshow(lowcontrast);

% Ensure the image is in uint8 format for histogram calculation
if ~isa(lowcontrast, 'uint8')
    lowcontrast = im2uint8(lowcontrast);
end

% Flatten the image to get pixel values as a one-dimensional array
pixel_values = lowcontrast(:);

% If pixel_values is not double, convert it to double
if ~isa(pixel_values, 'double')
    pixel_values = double(pixel_values);
end

% Display histogram with 256 bins
figure;
hist(pixel_values, 256);
title('Histogram of Original Image');

% Use histeq to enhance contrast using histogram equalization
lowcontrast_eq = histeq(lowcontrast);

% Display the enhanced image
figure;
imshow(lowcontrast_eq);
title('Enhanced Contrast Image');

% Display histogram of the enhanced image
figure;
pixel_values_eq = lowcontrast_eq(:);
hist(pixel_values_eq, 256);
title('Histogram of Enhanced Image');
```




Histogram of Original Image
1800
1600
1400
1200
1000
800
600
400
200
0
100
150
200
250

```
%% Q)3

% a) Add salt-and-pepper noise to the image with a noise density of
0.05
noise_density = 0.05;
lena_noisy = imnoise(lena_gray, 'salt & pepper', noise_density);
figure, imshow(lena_noisy), title('Noisy Image with Salt and Pepper
Noise');

% b) Filter the noise using the function medfilt2 with the 3x3 window
lena_filtered_3x3 = medfilt2(lena_noisy, [3 3]);
figure, imshow(lena_filtered_3x3), title('Image with 3x3 Median
Filter');

% c) Filter the noise with the 5x5 window
lena_filtered_5x5 = medfilt2(lena_noisy, [5 5]);
figure, imshow(lena_filtered_5x5), title('Image with 5x5 Median
Filter');
```