```
%% Image Interpolation Using Nearest neighbor technique


g=imread("im004.jpg");
% by imread command we can read the image in the form of 2-D array.
g=im2gray(g);
% converting colour Image to grayscale image
[r c]=size(g);
% taking the size of image or array

t=[];
% creating an array which will be used as final outpout of image.

temp=1;
% first temperory variable for visiting array elements

gemp=1;
% second temperory variable for visiting array elements

for i=1:1:r

    for j=1:1:c

        t(temp,gemp)=g(i,j); % Up sampling

        gemp=gemp+2;

    end

    temp=temp+2;

    gemp=1;

end

img_resized=uint8(t);
% storing the Upsampled image in new variable

figure;

imshow(g);
% displaying the Original Image

title('Original Image')

truesize
```

```

figure;

imshow(img_resized)
% Displaying the Upsampled image

title('Resized Image')
```

```
truesize
```

```
[R C] = size(t);

for i=1:2:R   % applying the Nearest neighbor Technique

    for j=1:2:C

        t(i,j+1)=t(i,j);
        % using the nearest neighbor pixel value to fill the empty pixels
        t(i+1,j)=t(i,j);
        % using the nearest neighbor pixel value to fill the empty pixels
        t(i+1,j+1)=t(i,j);
        % using the nearest neighbor pixel value to fill the empty pixels

    end

end


img_resized1=uint8(t);
% Storing the Transformed Image



figure;

imshow(img_resized1)
% Displaying the Tranformed image

title('Transformed Image')

truesize
```