PROJECT DOCUMENTATION

JavaFXProj (Top most file)- This class implements Load button for selecting images.
Button action is implemented in the class ‘OpenImage’
Dependent class- OpenImage

OpenImage- This class uploads the image as Thumbnails of size (100*100) and displays number of images selected by user.
Progress indicator will be displayed while images are being uploaded.
Back button is implemented if a user wants to cancel current selection and choose new images.
If a user does not want to select any image, cancel button in the dialogue box will terminate the program successfully.
Dependent classes- Tags, ImgUtils, DisplayProperty, ImageConversions

ImgUtils- This class provide encapsulation for the fields ‘image’, ‘SelectedFile’, and ‘format’.
The function ‘createProgressIndicator’ implements logic for notifying users that image is being uploaded.
createdir() function creates the *Downloaded_Images* directory relative to the upload path.
createtempdir() function creates a temporary directory *FilterTemp* relative to the upload path.
deleteDirectory(File dir): Deletes each file and the containing input directory, used to remove the temporary *FilterTemp* directory created.

Tags- This class is used to tag the filtered state of a file and is shown in the downloaded file name. It also contains a SelectedFile variable that is used to update the reference file after every filter is applied.

DisplayProperty- This class displays image properties. Image properties like image type, width, height, location are extracted from original image. Camera property is captured from EXIF data present in original image.
Dependent classes- ImgUtils

ImageFormat(Interface)- This Interface contains an abstract function ‘format()’ which is implemented by concrete classes 'BmpImage', 'GifImage', 'PngImage' and 'JpgImage'.

Formats (Inheritance)- This class contains logic for converting images into different format like jpg, bmp, gif, and png. This class is extended by concrete classes 'BmpImage', 'GifImage', 'PngImage' and 'JpgImage'.
Dependent classes- ImgUtils

BmpImage- This class implements the abstract function format() present in the interface ‘ImageFormat’. Implementation is achieved by extending with class ‘Format’.
Dependent classes- Formats, and ImageFormat

GifImage- This class implements the abstract function format() present in the interface ‘ImageFormat’. Implementation is achieved by extending with class ‘Format’.
Dependent classes- Formats, and ImageFormat

PngImage- This class implements the abstract function format() present in the interface ‘ImageFormat’. Implementation is achieved by extending with class ‘Format’.
Dependent classes- Formats, and ImageFormat

JpgImage- This class implements the abstract function format() present in the interface ‘ImageFormat’. Implementation is achieved by extending with class ‘Format’.
Dependent classes- Formats, and ImageFormat

ImageFormatFactory- This class creates and returns object of class 'BmpImage', 'GifImage', 'PngImage' and 'JpgImage' based on user request.

ImageFilter (Interface)- This Interface contains an abstract function filter()’ which is implemented by concrete class LogColorScale, GrayScale, CharcoalScale, ImplodeFilter, and OriginalImageFilter.

CharcoalScale- This class implements the abstract function filter() present in the interface ‘ImageFilter’. This function creates a Charcoal filtered version of the image and returns the view. The filepointer state is updated to the Charcoal scaled image and the tag to _charcoal_.

Dependent classes- Tags and ImgUtils

LogColorScale- This class implements the abstract function filter() present in the interface ‘ImageFilter’. This function creates a LogColorscale filtered version of the image and returns the view. The filepointer state is updated to the Logcolor scaled image and the tag to _logfiltered_.
Dependent classes- Tags and ImgUtils

GrayScale- This class implements the abstract function filter() present in the interface ‘ImageFilter’. This function creates a GrayScale filtered version of the image and returns the view. The filepointer state is updated to the Gray scaled image and the tag to _grayscale_.
Dependent classes- Tags and ImgUtils

ImplodeFilter- This class implements the abstract function filter() present in the interface ‘ImageFilter’. This function creates a imploded/exploded version of the image and returns the view. The filepointer state is updated to the implode filtered image and the tag to _implode_.
Dependent classes- Tags and ImgUtils

OriginalImageFilter- This class implements the abstract function filter() present in the interface ‘ImageFilter’. This function reproduces the original image without the effects of any filter and returns the view. The filepointer state is updated to the original image and the tag to _original_.
Dependent classes- Tags and ImgUtils

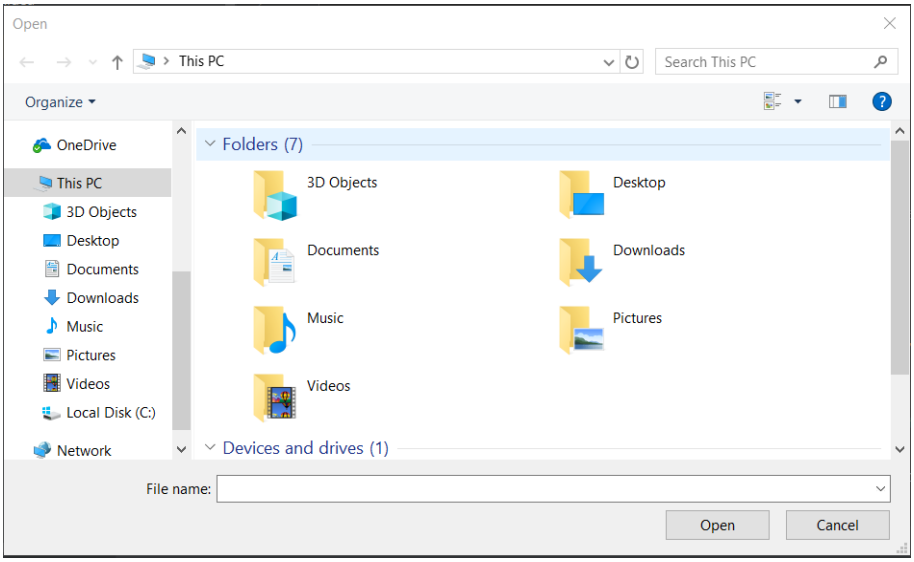
ImageFilterFactory- This class creates and returns object of class ‘LogColorScale’, ‘GrayScale’, ‘CharcoalScale’, ‘ImplodeFilter’, and ‘OriginalImageFilter’ based on user request.

ImageConversions- This class allows users to apply filters and download images in different formats. Filter button allows users to apply LogColorScale, GrayScale, CharcoalScale, ImplodeFilter, and OriginalImageFilter. Download button allows users to download images in different formats like png, jpg, bmp, and gif. Factory design pattern is used for creating objects of class ‘LogColorScale’, ‘GrayScale’, ‘CharcoalScale’, ‘ImplodeFilter’, ‘OriginalImageFilter’,
'BmpImage', 'GifImage', 'PngImage' and 'JpgImage'.
Dependent classes- ImgUtils, ImageFormatFactory, and ImageFilterFactory

JAVA IMAGE PROJECT TEST CASES

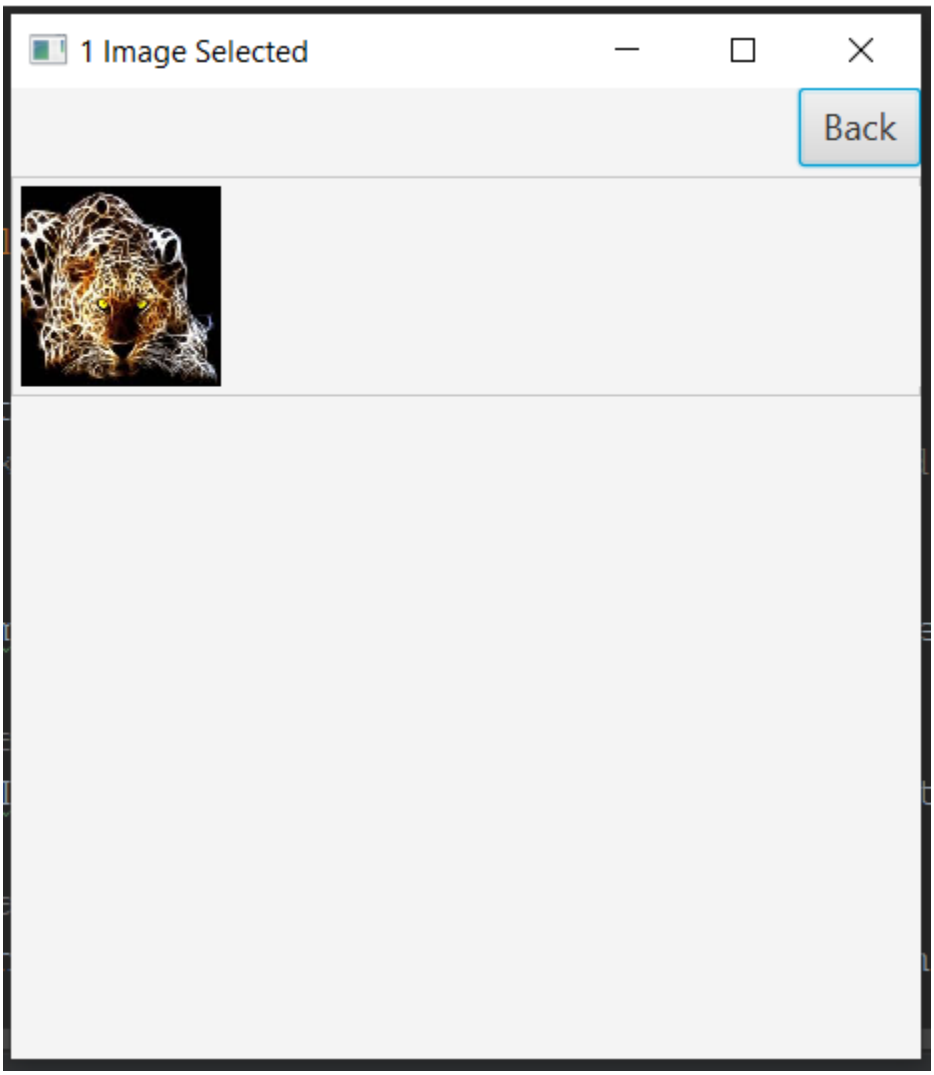
1. **Load button for selecting image**

2. **When select button is clicked, Dialogue box opens**

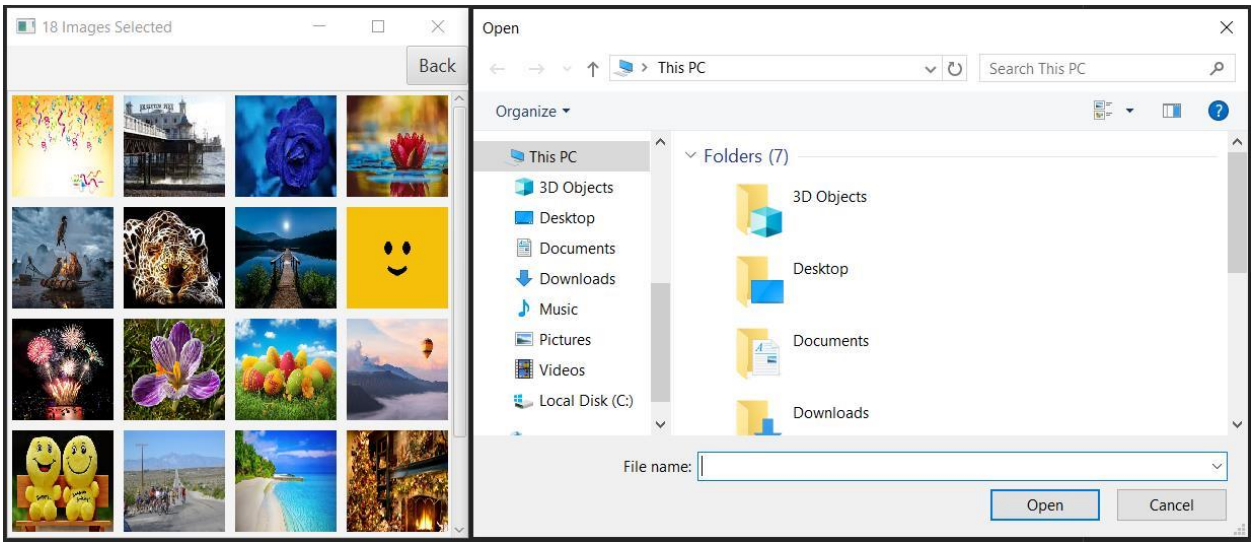


3. If a user clicks cancel button after opening dialog box , program terminates successfully

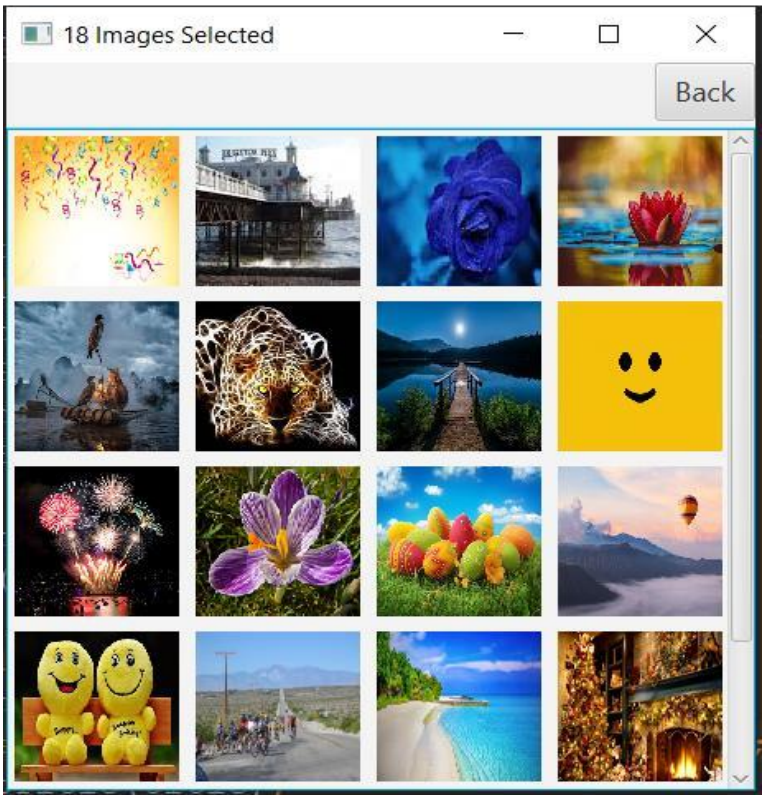
4. Single image selection by user



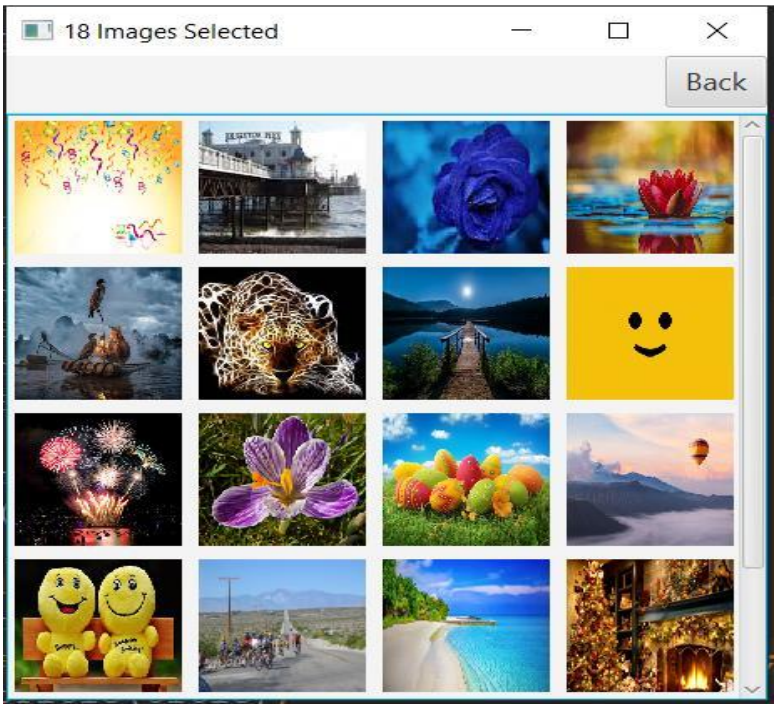
5. Back button for cancelling current selections and choosing new images



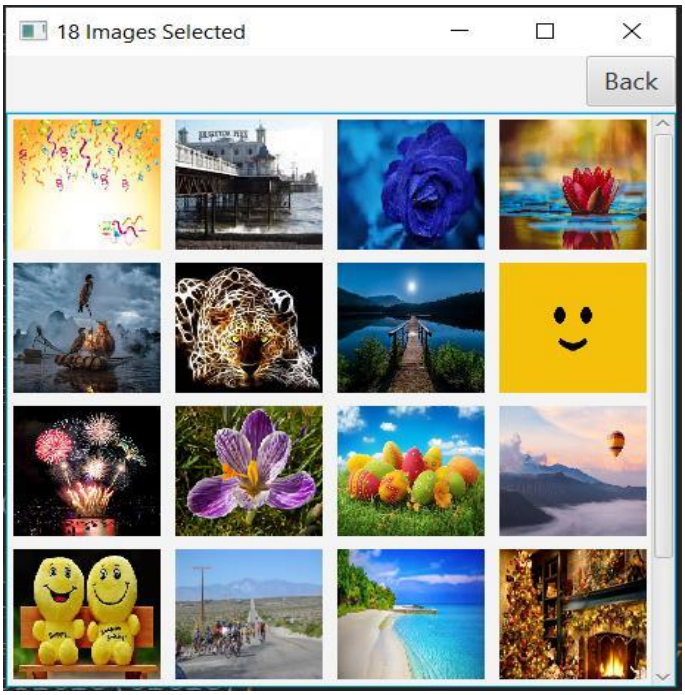
6. Multiple image selection by user and display them as thumbnails.



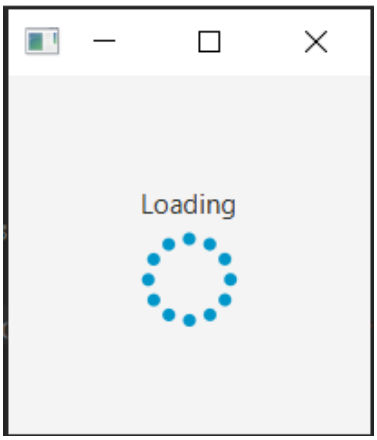
7. Displaying number of images selected by user.



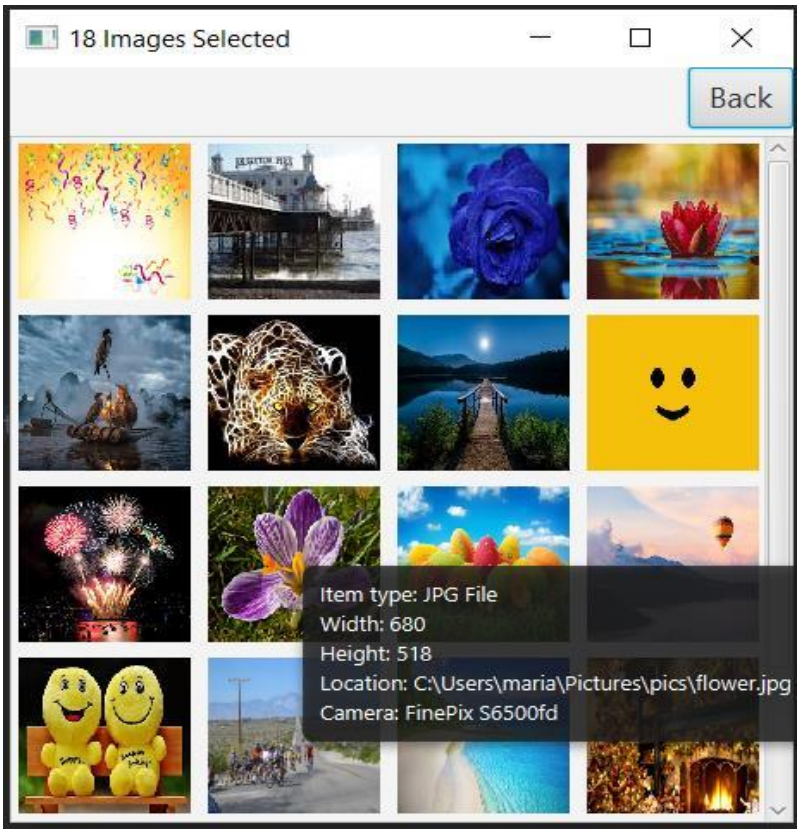
8. Display Scroll bar if more than 16 images are selected



9. Progress indicator while loading images

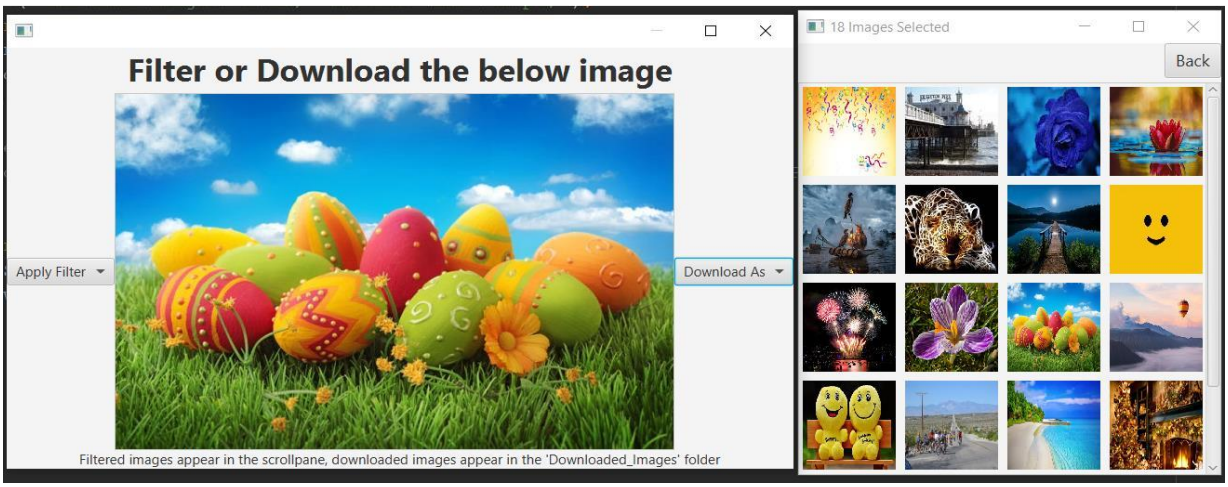


10. Tooltip for displaying image properties (type, height, width, camera (Image should have EXIF data for getting camera information) and location)



11. If user clicks close button after selecting images, program exit successfully

12. When a second stage (stage with filter and download option) is opened, the primary stage (stage on the right side of below image) is made inactive to prevent multiple stage opening on the same screen.

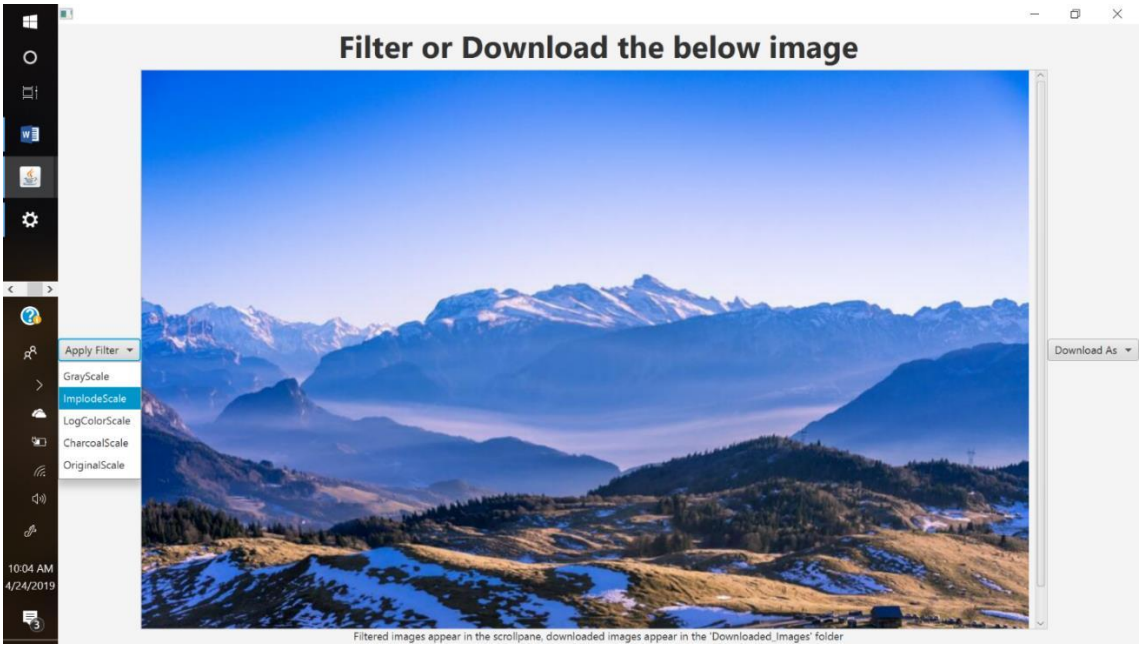


13. Gray Scale Filter:

Transforms the original image into a gray scale filtered image, and places it in the scrollpane

14. Implode Scale Filter:

Transforms the original image into an image imploded form the center, and places it in the scrollpane



15. Difference between Implode Filter vs Original Image:

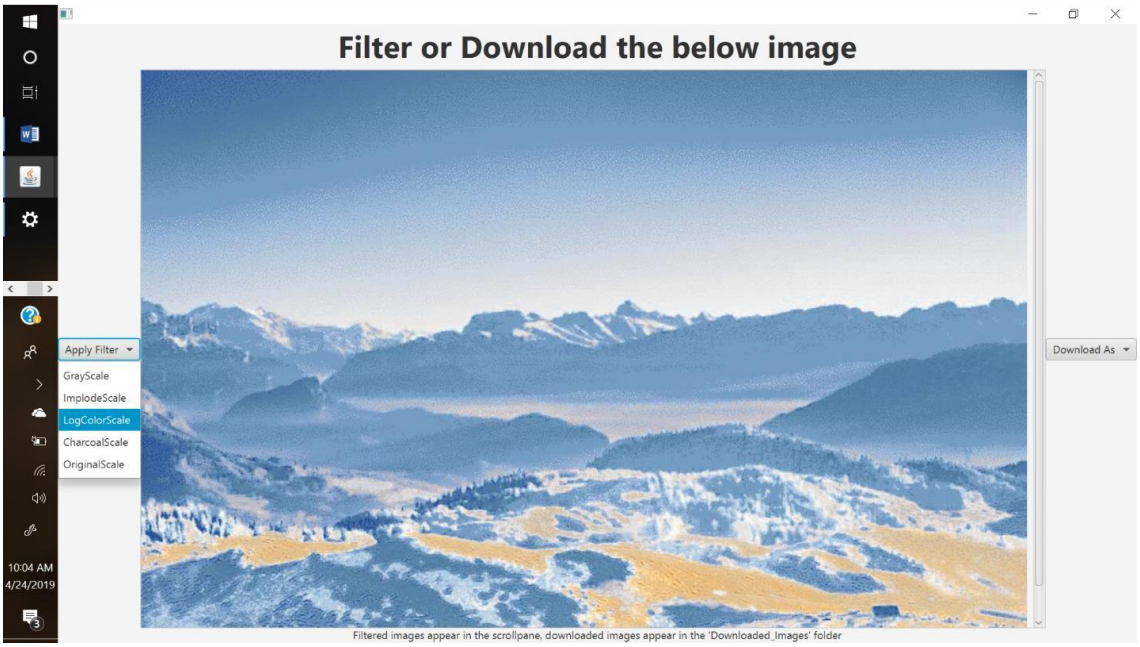
Transforms the image from the center as seen below, it is easy to notice the difference in the scenary use case below:

ORIGINAL

IMPLODE FILTERED

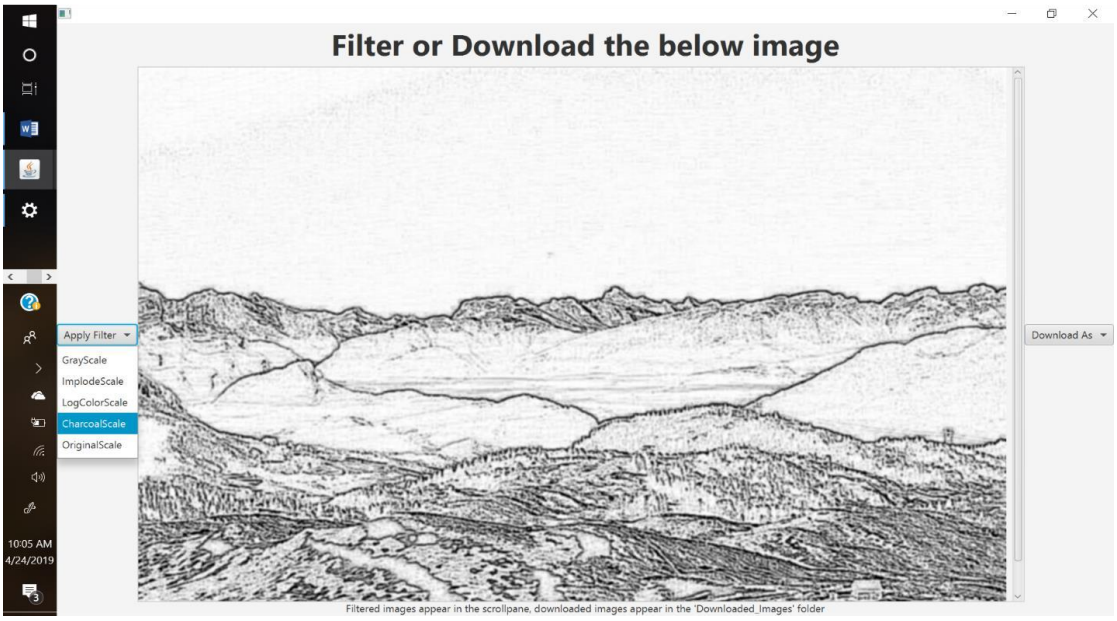
16. Log Color Scale Filter:

Transforms the original image into a logcolor filtered image, and places it in the scrollpane



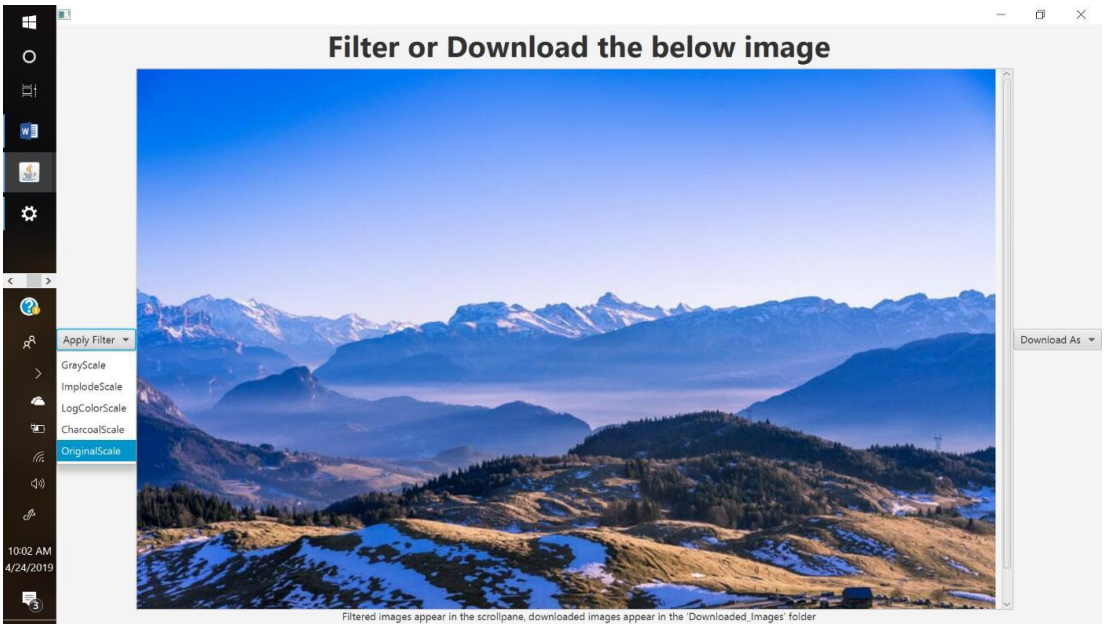
17. Charcoal Scale Filter:

Transforms the original image into a charcoal filtered image, and places it in the scrollpane



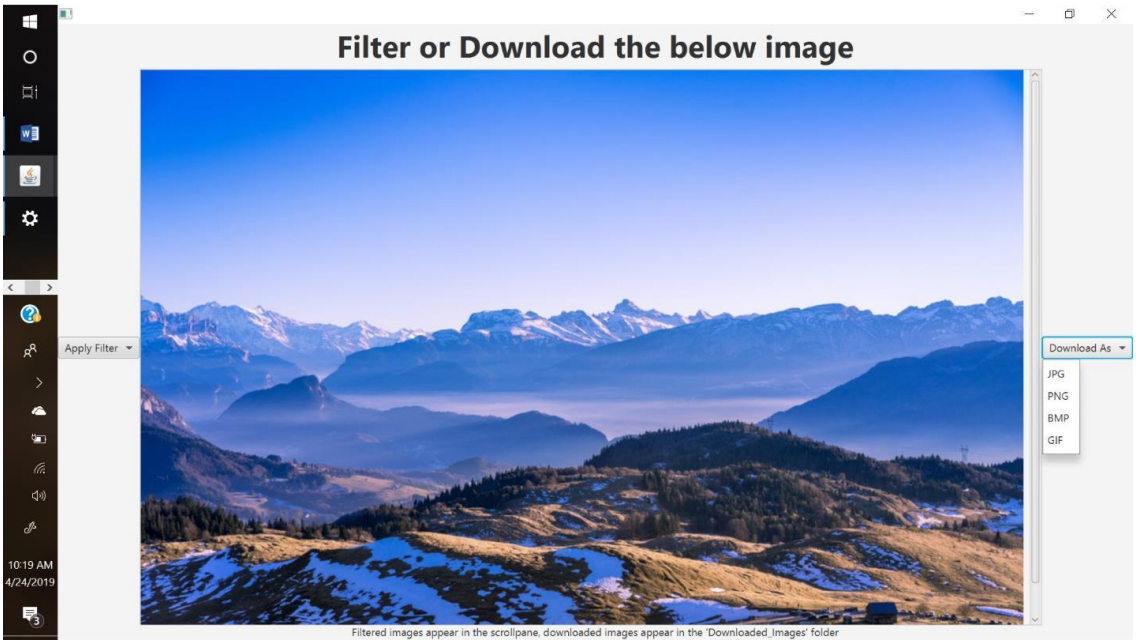
18. Original Scale:

Reverts the scrollpane filtered image into the original image, screenshot below by applying Original Scale on a charcoal filtered image:



19. Download Tests:

Download formats available in JPG, PNG, BMP, and GIF as shown below in the Download As dropdown



- **The downloaded images are available in the Download Images folder relative to the path from which the images were uploaded.**

- **Example download use cases for both charcoal filtered and implode filtered images were conducted. These filtered images were downloaded in all the available formats (JPG, PNG, BMP, GIF) and the results are shown below:**