Exercise 1: plotting an image from a FITS file

Using NumPy, Matplotlib, and Astropy, read in the file `lmc_ha_b20.fits` and do the following:

- Get the image data from the first HDU.

In [11]:

```
#Code Here
from astropy.io import fits
import numpy as np
path = r"C:\Users\eklav\OneDrive - University of Illinois - Urbana\astro_310\labs\l
image = fits.open(path)
image
image_data = image[0].data
image_data
```

Out[11]:

```
array([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]], dtype='>f4')
```

- Take the common logarithm of the image data.

In [12]:

```
#Code Here
log_image = np.log(np.where(image_data > 0, image_data, np.nan))
log_image
```

Out[12]:

```
array([[nan, nan, nan, ..., nan, nan, nan],
       [nan, nan, nan, ..., nan, nan, nan],
       [nan, nan, nan, ..., nan, nan, nan],
       ...,
       [nan, nan, nan, ..., nan, nan, nan],
       [nan, nan, nan, ..., nan, nan, nan],
       [nan, nan, nan, ..., nan, nan, nan]], dtype=float32)
```

- Get the first HDU's keywords `NAXIS1`, `NAXIS2`, `CDELT1`, and `CDELT2` (the latter two are in degrees/pixel). Use this information to compute the extent of the image. (Use the abs value of `CDELT1` ... RA goes backwards)

In [14]:

```
#Code Here
naxis1 = image[0].header['NAXIS1']
naxis2 = image[0].header['NAXIS2']
cdelt1 = image[0].header['CDELT1']
```

```
cdelt2 = image[0].header['CDELT2']
image.close()
```

- Plot the log image using the extent determined by the above keywords and placing the origin at lower left.

In [16]:
```
#Code Here
extent = [0, naxis1 * abs(cdelt1), 0, naxis2 * abs(cdelt2)]
print("log image data :", log_image)
print("naxis1", naxis1)
print("\naxis2", naxis2)
print("\ncdelt1", cdelt1)
print("\ncdelt2", cdelt2)
print("\nextent", extent)
# QUITE OBVIOUSLY MOST OF THE DATA WOULD HAVE BEEN 0s since its an image which was
```

```
log image data : [[nan nan nan ... nan nan nan]
 [nan nan nan ... nan nan nan]
 [nan nan nan ... nan nan nan]
 ...
 [nan nan nan ... nan nan nan]
 [nan nan nan ... nan nan nan]
 [nan nan nan ... nan nan nan]]
naxis1 540

axis2 540

cdelt1 -0.0166666664183139

cdelt2 0.016666666418314

extent [0, 8.999999865889507, 0, 8.999999865889562]
```

Exercise 2: modifying a FITS table

- The files `table1.fits` and `table2.fits` each contain an empty primary HDU and a binary table extension HDU. The table data are drawn from a sample of nearby clusters of galaxies.
- The file `table1.fits` contains: cluster name, redshift, RA (J2000) and dec (J2000).
- The file `table2.fits` contains: cluster name, 2-10 keV X-ray luminosity, X-ray temperature in keV.
- Using NumPy and Astropy, read the two files in, merge the tables into one new table, and write the results to a new file `newtable.fits`.

In [26]:
```
#Code Here
from astropy.table import Table, join
```

```
table1 = Table.read('table1.fits', hdu=1)
table2 = Table.read('table2.fits', hdu=1)
# merge = join(table1, table2, keys='cluster name')
# cluster name does not exist only name is present
merge = join(table1, table2, keys='name')
merge.write('newtable.fits', format='fits', overwrite=True)
print(table1.columns)
print(table2.columns)
```

```
<TableColumns names=('name','redshift','raj2000','decj2000')>
<TableColumns names=('name','lx2-10','tx')>
```