FITS, The Astronomy Data

Since, we know the astronomical data collected by a any typical computational telescope is very vast, often the data is stored as **.fits** (Flexible Image Transport System), the data in the fits format is formatted as multidimensional data(2D array)

Reading the data using **Astropy**
from astropy import fits

```
In [45]:
header = fits.open('image0.fits')
print('meta data :\n')
header.info()
```

```
meta data :

Filename: image0.fits
No.    Name      Ver    Type      Cards   Dimensions   Format
  0  PRIMARY       1 PrimaryHDU       7   (200, 200)   float64
```

fits.open('filename.fits')

fits file contains a header where the metadata is stored, Metadata is nothing but a brief information about the data and the way its formatted

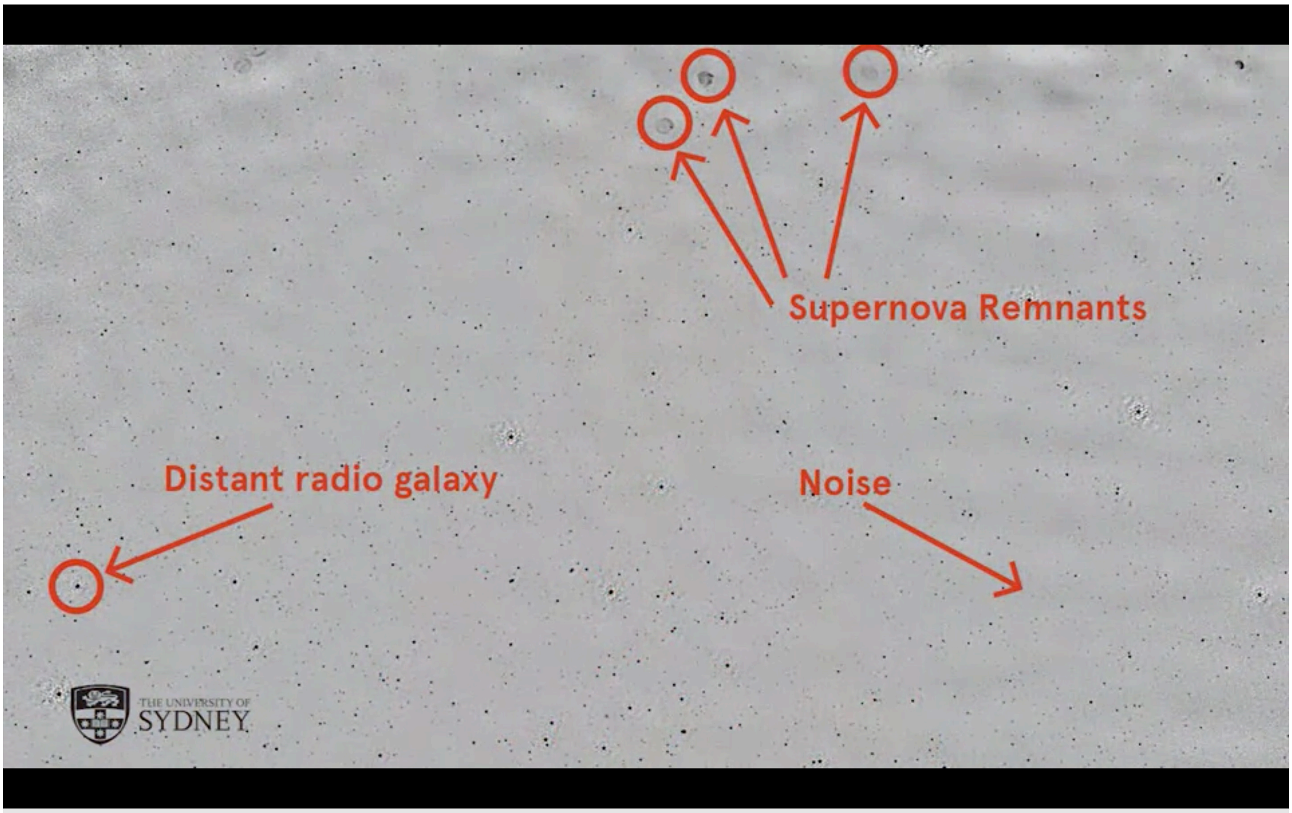
Significance of the Data points

```
In [48]:
header = fits.open('image0.fits')
print('Data shape:',header[0].data.shape)
header[0].data
```

```
Data shape: (200, 200)

array([[-0.00319621, -0.0008495 ,  0.00241029, ..., -0.00395876,
        -0.00044348, -0.00068918],
       [ 0.00047027, -0.00116831, -0.00097984, ..., -0.0025309 ,
        -0.0029578 ,  0.00056235],
       [-0.00398995, -0.0014535 , -0.00069149, ..., -0.00140105,
        -0.00352348, -0.00031844],
       ...,
       [-0.0040899 , -0.00576512, -0.00548605, ..., -0.00326335,
        -0.00336867,  0.00071865],
       [-0.00648206, -0.0040749 , -0.00342054, ..., -0.00481083,
        -0.00295951, -0.00204938],
       [-0.00437305, -0.00450521, -0.00194182, ..., -0.00775742,
        -0.00532724, -0.00222068]])
```

In our case, fits file contain the Image data encoded numerically, each point signifies as a pixel of an Image taken by the MWA telescope.



?This is an Example image taken by the MWA telescope, the grey scale background is considered as Noise, and the dark spots are either distant galaxies or supernova remnants as shown.
So, if we find out the max value of a pixel in the image.fits it can either be a supernova remnant or a Distant radio galaxy

Task one is loading the fits image we've already know that,
Task two is to find the point where the brightest possible pixel is located in the image density. which can be done by
np.unravel_index(np.argmax(data, axis=None), data.shape)
np.uravel_index explanation

Plotting..

```
# Run your `load_fits` function with examples:
  bright = load_fits('image0.fits')
  print(bright)

  hdulist = fits.open('image0.fits')
  data = hdulist[0].data

# Plot the 2D image data
  plt.figure(figsize=(7,7))
  plt.xlabel('x-pixels (RA)',size=20)
  plt.ylabel('y-pixels (dec)',size=20)
  plt.title('Supernova remnants',size=20)
  plt.annotate(bright,
  xytext = (100, 75),
  xy = bright,
  arrowprops = { 'facecolor' : 'grey', 'shrink' : 0.1 })
  plt.imshow(data.T, cmap=plt.cm.inferno)
  plt.colorbar()
  plt.show()
```

loads_fits() function returns the point of brightest pixel, each data point is equal to a pixel, matplotlib has a plot type which is used to plot numerical data plt.imshow()

the cmap argument in the imshow is to choose the type of gradient to fill the plot

plt.annotate on the other hand is optional an i used it to mention a arrow on the brightest pixel

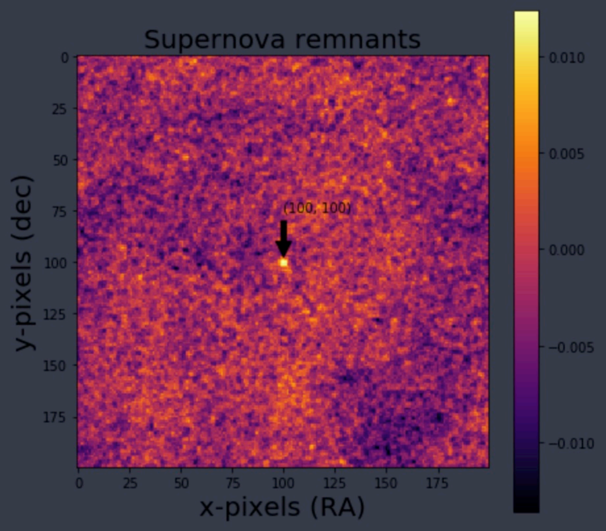
plt.annotate('name',xytext=(),xy=point,arrowprops={'facecolor':'red','shrink':0.1})

xytext = should have a point as argument which lies in the plots default or manually defined xlim and ylim

xy should be the point where the arrow points

The plot

```
The brighest point is (100, 100)
```




```
This point now can either be a potential pulsar detection or a distant radio galaxy, which is determined after further examinat
ion of the data
```

After this we use multiple fit files to take consider the mean of the data and then plot the data