File: HWK7.ipynb

Description:

Name: Dyab Asdi

UT EID: da32435

In [153…

```
import numpy as np
import pandas as pd
```

In [154…

```
df=pd.read_csv('./bsc5.csv')
```

In [155…

```
df.shape
```

Out[155…

```
(9110, 53)
```

In [156…

```
df.head(5)
```

Out[156…

	HR	Name	DM	HD	SAO	FK5	IRflag	r_IRflag	Multiple	ADS	...	RadVel	n_Rac
0	1	NaN	BD+44 4550	3.0	36042.0	NaN	NaN	NaN	NaN	46.0	...	-18.0	N
1	2	NaN	BD-01 4525	6.0	128569.0	NaN	NaN	NaN	NaN	NaN	...	14.0	
2	3	33 Psc	BD-06 6357	28.0	128572.0	1002.0	I	NaN	NaN	NaN	...	-6.0	SI
3	4	86 Peg	BD+12 5063	87.0	91701.0	2004.0	NaN	NaN	NaN	NaN	...	-2.0	
4	5	NaN	BD+57 2865	123.0	21085.0	NaN	NaN	NaN	NaN	61.0	...	-12.0	

5 rows × 53 columns

In [157…

```
df.tail(5)
```

Out[157…

	HR	Name	DM	HD	SAO	FK5	IRflag	r_IRflag	Multiple	ADS	...	RadVe
9105	9106	NaN	CP-73 2346	225233.0	255629.0	NaN	NaN	NaN	NaN	NaN	...	8.
9106	9107	NaN	BD+33 4828	225239.0	53622.0	2002.0	NaN	NaN	NaN	NaN	...	4.
9107	9108	NaN	CP-72 2800	225253.0	255631.0	1001.0	NaN	NaN	NaN	NaN	...	-3.
9108	9109	NaN	BD+25 5068	225276.0	73731.0	NaN	I	NaN	NaN	42.0	...	-5.
9109	9110	NaN	BD+60 2667	225289.0	10962.0	NaN	NaN	NaN	NaN	NaN	...	14.

5 rows × 53 columns

In [158…

```
df.describe()
```

Out[158…

	HR	HD	SAO	FK5	ADS	RAh1900	RAm1
count	9110.00000	9096.000000	9071.000000	3077.000000	1435.000000	9096.000000	9096.00
mean	4555.50000	107334.581794	134193.509536	1954.447514	8278.804878	11.555739	29.66
std	2629.97481	67739.321279	75516.511021	1209.016612	4700.699670	6.787416	17.18
min	1.00000	3.000000	106.000000	1.000000	1.000000	0.000000	0.00
25%	2278.25000	44330.500000	71659.000000	820.000000	4154.500000	6.000000	15.00
50%	4555.50000	103449.500000	131443.000000	2104.000000	8406.000000	11.000000	30.00
75%	6832.75000	167648.750000	200666.500000	3046.000000	11822.500000	18.000000	45.00
max	9110.00000	225289.000000	258996.000000	3997.000000	17175.000000	23.000000	59.00

8 rows × 31 columns

In [159…

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9110 entries, 0 to 9109
Data columns (total 53 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   HR          9110 non-null   int64  
 1   Name        3157 non-null   object 
 2   DM          9096 non-null   object 
 3   HD          9096 non-null   float64
 4   SAO         9071 non-null   float64
 5   FK5         3077 non-null   float64
 6   IRflag      1743 non-null   object 
 7   r_IRflag    84 non-null     object 
 8   Multiple    1577 non-null   object 
 9   ADS         1435 non-null   float64
 10  ADScomp     403 non-null    object 
 11  VarID       2182 non-null   object 
 12  RAh1900     9096 non-null   float64
 13  RAm1900     9096 non-null   float64
 14  RAs1900     9096 non-null   float64
 15  DE-1900     9096 non-null   object 
 16  DEd1900     9096 non-null   float64
 17  DEm1900     9096 non-null   float64
 18  DEs1900     9096 non-null   float64
 19  RAh         9096 non-null   float64
 20  RAm         9096 non-null   float64
 21  RAs         9096 non-null   float64
 22  DE-         9096 non-null   object 
 23  DEd         9096 non-null   float64
 24  DEm         9096 non-null   float64
 25  DEs         9096 non-null   float64
 26  GLON        9096 non-null   float64
 27  GLAT        9096 non-null   float64
 28  Vmag        9096 non-null   float64
 29  n_Vmag      306 non-null    object 
 30  u_Vmag      1 non-null      object 
 31  B-V         8786 non-null   float64
 32  u_B-V       30 non-null     object 
 33  U-B         7206 non-null   float64
 34  u_U-B       46 non-null     object 
 35  R-I         2551 non-null   float64
 36  n_R-I       303 non-null    object 
 37  SpType      9096 non-null   object 
 38  n_SpType    427 non-null    object 
 39  pmRA        9096 non-null   float64
 40  pmDE        9096 non-null   float64
 41  n_Parallax  388 non-null    object 
 42  Parallax    3289 non-null   float64
 43  RadVel      9092 non-null   float64
 44  n_RadVel    4958 non-null   object 
 45  l_RotVel    724 non-null    object 
 46  RotVel      3933 non-null   float64
 47  u_RotVel    284 non-null    object 
 48  Dmag        2890 non-null   float64
 49  Sep         2951 non-null   float64
 50  MultID      1114 non-null   object 
 51  MultCnt     1148 non-null   float64
 52  NoteFlag    4477 non-null   object 
dtypes: float64(30), int64(1), object(22)
memory usage: 3.7+ MB
```

In [160…

```
# data cleaning
df.dropna(subset = ['RAh'], axis = 0, inplace = True)
df.reset_index(drop=True, inplace=True)
df=df.astype({'RAh':int, 'RAm':int,'RAs':float})
df=df.astype({'DEd':int, 'DEm':int, 'DEs':float})
df=df.astype({'GLON':float, 'GLAT':float})
```

In [161…

```
# RA and Dec to degrees conversion
df['ra_deg'] = 15 * (df['RAh'] + df['RAm']/60 + df['RAs']/3600)
df['dec_deg'] = np.abs(df['DEd']) + df['DEm']/60 + df['DEs']/3600
df.loc[df['DE-'] == '-', 'dec_deg'] *= -1
```

In [162…

```
#convert right ascension to degrees
df['ra_deg'] = 15 * (df['RAh'] + df['RAm'] / 60.0 + df['RAs'] / 3600.00)
#convert declination to degrees
df['dec_deg'] = df['DEd'] + df['DEm'] / 60.0 + df['DEs'] / 3600.0
df = df.astype({'dec_deg':str})
df['dec_deg'] = df['DE-'] + df['dec_deg']
df=df.astype({'dec_deg':float})
df.head()
```

Out[162…

	HR	Name	DM	HD	SAO	FK5	IRflag	r_IRflag	Multiple	ADS	...	l_RotVel	Rot\
0	1	NaN	BD+44 4550	3.0	36042.0	NaN	NaN	NaN	NaN	46.0	...	NaN	19!
1	2	NaN	BD-01 4525	6.0	128569.0	NaN	NaN	NaN	NaN	NaN	...	NaN	Na
2	3	33 Psc	BD-06 6357	28.0	128572.0	1002.0	I	NaN	NaN	NaN	...	<	17
3	4	86 Peg	BD+12 5063	87.0	91701.0	2004.0	NaN	NaN	NaN	NaN	...	NaN	Na
4	5	NaN	BD+57 2865	123.0	21085.0	NaN	NaN	NaN	NaN	61.0	...	NaN	Na

5 rows × 55 columns

In [163…

```
new_df = df['n_SpType'].dropna()
print(new_df)
```

```
6       e
14      v
25      e
44      e
47      e
       ..
9061    v
9065    v
9074    e
9082    e
9088    v
Name: n_SpType, Length: 427, dtype: object
```

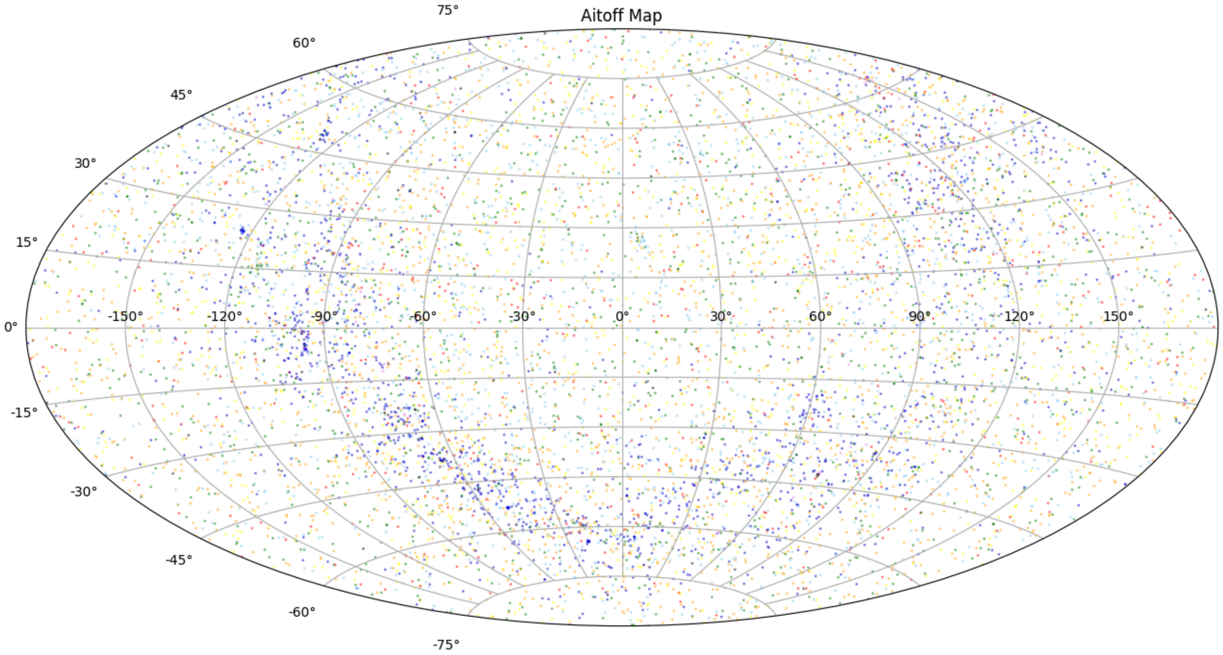
In [164…

```
# Define spectral type to color mapping
def get_star_color(spectral_type):
    if pd.isna(spectral_type):
        return 'black'  # Default for missing values
    spectral_type = spectral_type.strip().upper()  # Normalize input
    if spectral_type.startswith('O'):
        return 'violet'
    elif spectral_type.startswith('B'):
        return 'mediumblue'
    elif spectral_type.startswith('A'):
        return 'lightskyblue'
    elif spectral_type.startswith('F'):
        return 'green'
    elif spectral_type.startswith('G'):
        return 'yellow'
    elif spectral_type.startswith('K'):
        return 'orange'
    elif spectral_type.startswith('M'):
        return 'red'
    return 'black'  # Default for unknown types


# Apply function to create 'color' column
df['color'] = df['SpType'].apply(get_star_color)


print(df['color'])
```

```
0       lightskyblue
1             yellow
2             orange
3             yellow
4             yellow
            ...     
9091           green
9092          yellow
9093      mediumblue
9094          orange
9095      mediumblue
Name: color, Length: 9096, dtype: object
```

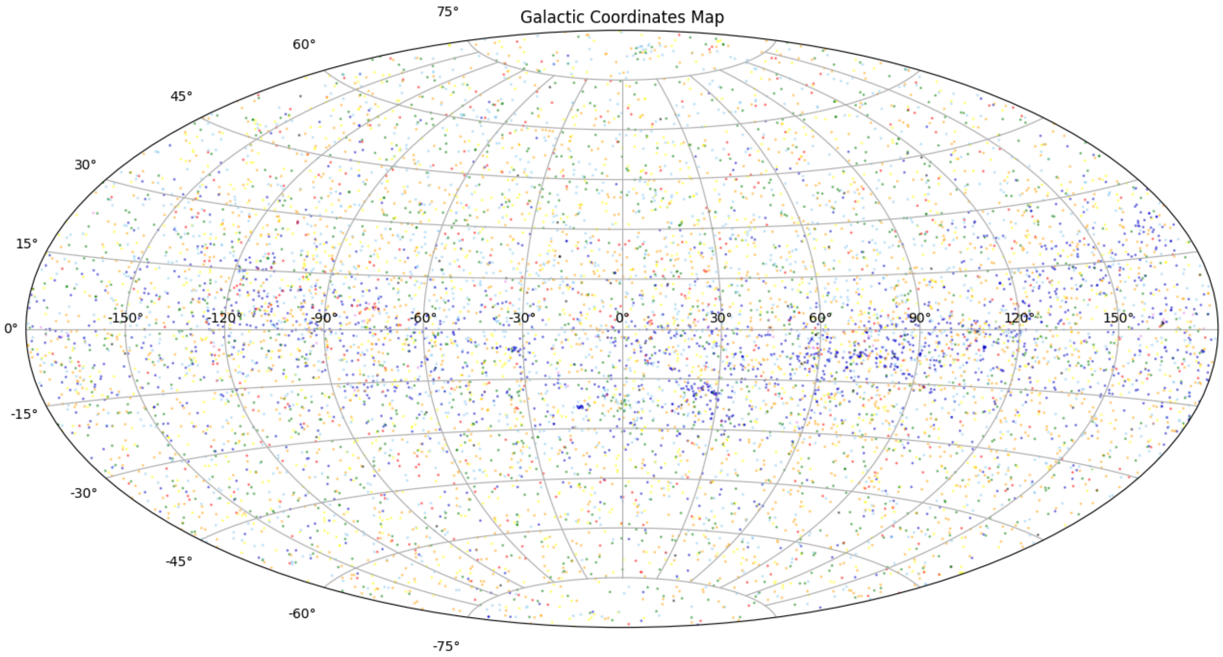
In [165…

```
import matplotlib.pyplot as plt
ra_rad = np.radians(df['ra_deg'] - 180)
dec_rad = np.radians(df['dec_deg'])
plt.figure (figsize = (16, 8.4))
plt.subplot (projection = "aitoff")
plt.title ("Aitoff Map")
plt.scatter(ra_rad, dec_rad, c=df['color'], s=1, alpha=0.4, marker='o')
plt.grid (True)
plt.show()
```



In [166…

```
# Conversion to radians
glon_rad = np.radians(df['GLON'] - 180)
glat_rad = np.radians(df['GLAT'])

# Creating plot
plt.figure(figsize=(16, 8.4))
plt.subplot(projection="aitoff")
plt.title("Galactic Coordinates Map")
plt.scatter(glon_rad, glat_rad, c=df['color'], s=1, alpha=0.4, marker='o')
plt.grid(True)
plt.show()
```



Equatorial:

The equatorial coordinates are aligned with the earth's rotation axis, and uses Right Ascension and Declination to plot position, with the main plane being the Celestial Equator.

Galactic:

The galactic coordinates are aligned with the milky way's center, with the main plane being the galactic plane. It uses galactic longitude and galactic latitude to plot.