Exercise 1: units and constants

Using Astropy, compute the following quantities. Hints on how to get started are in the reading for today!

1. The escape velocity (in km/s) of a $2.3 M_\odot$ giant star with a radius of $150 R_\odot$.

$$v_{esc} = \sqrt{\frac{2GM}{R}}$$

In [12]:
```
#Code Here

from astropy.constants import M_sun , R_sun, G , b_wien
import numpy as np
import astropy.units as u

v_esc =  np.sqrt(2 * M_sun * 2.3 * G / (R_sun * 150)).to(u.km/u.s)

print(v_esc)
```

```
76.48532239709606 km / s
```

2. The wavelength (in angstroms) of the peak of the spectrum of a $11,350 K$ blackbody.

$$\lambda_{peak} = \frac{0.29 \text{ cm K}}{T}$$

In [17]:
```
#Code Here
lambda_peak = (b_wien / (11350 * u.K)).to(u.Angstrom)
lambda_peak
```

Out[17]: $2553.103 \ \mathrm{\mathring{A}}$

3. The orbital period (in minutes) of a $1 M_\odot$ white dwarf orbiting a $0.6 M_\odot$ white dwarf at a separation of 10^5 km.

$$P = 2\pi\sqrt{\frac{a^3}{G(M_1 + M_2)}}$$

In [24]:
```
#Code Here
M1 = 1 * M_sun
M2 = 0.6 * M_sun
a = 10 **5 * u.km

P = (2 * np.pi * np.sqrt(a**3 / (G * (M1 + M2)))).to(u.min)
P
```

Out[24]: 7.1864235 min

4. The Compton wavelength (in fm) of a nickel-56 nucleus, given its rest mass of $52.110 GeV/c^2$.

$$\lambda = \frac{h}{mc}$$

In [41]:
```
#Code Here
import astropy.constants as const
# print(const.h.value) # this is is much better way than importing all one by one
mass_rest = 52.110 * u.GeV / (const.c**2)

mass_kg = mass_rest.to(u.kg, equivalencies=u.mass_energy())
wavelength = const.h / (mass_rest * const.c)

l_compton = wavelength.to(u.fm)

l_compton
```

Out[41]: 0.023792784 fm

5. The distance (in Mpc) to a radio galaxy with a flux density of $8400 Jy$ and a spectral luminosity of 6×10^{35} erg s^{-1} Hz^{-1}.

$$F_\nu = \frac{L_\nu}{4\pi d^2}$$

In [46]:
```
#Code Here

flux = 8400 * u.Jy
L_v = 6e35 * u.erg / (u.s * u.Hz)
flux_SI = flux.to(u.W / (u.m**2 * u.Hz))

L_v_SI = L_v.to(u.W / u.Hz)

dist = np.sqrt(L_v_SI /(4 * np.pi * flux_SI)).to(u.Mpc)

dist
```

Out[46]: 244.33209 Mpc

Exercise 2: sky coordinate conversions and angles

Using `SkyCoord`, `EarthLocation`, and `Time` objects, perform the following calculations:

- Find the angular distance between the Galactic center ($l = 0^\circ$, $b = 0^\circ$ in galactic coordinates) and the globular cluster M13.

In [51]:
```python
#Code Here
from astropy.coordinates import SkyCoord
galactic_center = SkyCoord(l=0*u.degree, b=0*u.degree, frame='galactic')

m13 = SkyCoord(ra=250.423475*u.degree, dec=36.461319*u.degree, frame='icrs')

m13_galactic = m13.transform_to('galactic')

angular_distance = galactic_center.separation(m13_galactic)

print("Amgular distance bw Galatic Center and M13 :", angular_distance)
```

```
Amgular distance bw Galatic Center and M13 : 67d06m01.56335465s
```

- Find the angular distance between Venus and the Sun on February 29, 2024 at 23:00 UTC (i.e. “2024-02-29T23:00:00”). Use the `astropy.coordinates.get_body()` method to get sky coordinates for Solar System bodies at a given time.

In [54]:
```python
#Code Here
from astropy.coordinates import get_body, solar_system_ephemeris
from astropy.time import Time

time = Time('2024-02-29T23:00:00')
venus = get_body('venus', time)
sun = get_body('sun', time)
ang_dist_v_s = venus.separation(sun)
print("Angular distance between Venus and the Sun :", ang_dist_v_s)
```

```
Angular distance between Venus and the Sun : 24d24m56.51528799s
```

- Create an altitude-azimuth coordinate frame object (`AltAz`) corresponding to the location of the ALMA Observatory at the same time as above. Create a `SkyCoord` object corresponding to the sky position of the Large Magellanic Cloud (RA 05h23m34.5s, Dec $-69^{\circ}45'22''$ in the `icrs` frame). Finally, using the `SkyCoord` object’s `transform_to` method, find its representation in the alt-az frame you created. This will give the altitude and azimuth of the LMC at ALMA this evening at 8 pm local time.

In [61]:
```python
#Code Here
from astropy.coordinates import EarthLocation, AltAz
alma_loc = EarthLocation(lat=-23.0262015*u.deg, lon=-67.7551257*u.deg, height=5060*

time = Time('2024-02-29T20:00:00') - alma_loc.lon.hourangle * u.hour
altaz_f = AltAz(obstime=time, location=alma_loc)
lmc_i = SkyCoord(ra='05h23m34.5s', dec='-69d45m22s', frame='icrs')
lmc_a = lmc_i.transform_to(altaz_f)
print("LMC at ALMA - Altitude :" , lmc_a.alt, "Azimuth:" ,lmc_a.az)
```

```
LMC at ALMA - Altitude : 42d00m36.51718381s Azimuth: 188d29m17.85090159s
```