Olympic Statistics Project Proposal

Step 1: Preparing for Your Proposal 1. Which client/dataset did you select and why?

I chose SportsStats (Olympics Dataset - 120 years of data). Because I discovered an intriguing Olympic statistic

2. Describe the steps you took to import and clean the data.

I use Pandas to import a csv file into the Jupiter notebook.

```
In [5]: import numpy as np
        import pandas as pd

In [9]: athlete_events=pd.read_csv('athlete_events.csv')

In [10]: noc_regions=pd.read_csv('noc_regions.csv')

In [11]: athlete_events.head
```

```
Out[11]: <bound method NDFrame.head of          ID                Name Sex  Age  Height  Weight \
0         1             A Dijiang   M  24.0   180.0    80.0
1         2             A Lamusi    M  23.0   170.0    60.0
2         3      Gunnar Nielsen Aaby   M  24.0     NaN     NaN
3         4      Edgar Lindenau Aabye   M  34.0     NaN     NaN
4         5 Christine Jacoba Aaftink   F  21.0   185.0    82.0
...     ...                ...  ..  ...     ...     ...
271111  135569            Andrzej ya   M  29.0   179.0    89.0
271112  135570              Piotr ya   M  27.0   176.0    59.0
271113  135570              Piotr ya   M  27.0   176.0    59.0
271114  135571      Tomasz Ireneusz ya   M  30.0   185.0    96.0
271115  135571      Tomasz Ireneusz ya   M  34.0   185.0    96.0

              Team  NOC        Games  Year  Season          City \
0            China  CHN  1992 Summer  1992  Summer     Barcelona
1            China  CHN  2012 Summer  2012  Summer        London
2          Denmark  DEN  1920 Summer  1920  Summer      Antwerpen
3      Denmark/Sweden  DEN  1900 Summer  1900  Summer         Paris
4      Netherlands  NED  1988 Winter  1988  Winter       Calgary
...            ...  ...          ...   ...     ...           ...
271111    Poland-1  POL  1976 Winter  1976  Winter     Innsbruck
271112      Poland  POL  2014 Winter  2014  Winter        Sochi
271113      Poland  POL  2014 Winter  2014  Winter        Sochi
271114      Poland  POL  1998 Winter  1998  Winter       Nagano
271115      Poland  POL  2002 Winter  2002  Winter  Salt Lake City

             Sport                        Event Medal
0       Basketball       Basketball Men's Basketball   NaN
1             Judo   Judo Men's Extra-Lightweight   NaN
2         Football         Football Men's Football   NaN
3       Tug-Of-War     Tug-Of-War Men's Tug-Of-War  Gold
4     Speed Skating  Speed Skating Women's 500 metres   NaN
...             ...                              ...   ...
271111        Luge        Luge Mixed (Men)'s Doubles   NaN
271112  Ski Jumping  Ski Jumping Men's Large Hill, Individual   NaN
```

```
In [19]: athlete_events.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 271116 entries, 0 to 271115
Data columns (total 15 columns):
 #   Column  Non-Null Count   Dtype
---  ------  --------------   -----
 0   ID      271116 non-null  int64
 1   Name    271116 non-null  object
 2   Sex     271116 non-null  object
 3   Age     261642 non-null  float64
 4   Height  210945 non-null  float64
 5   Weight  208241 non-null  float64
 6   Team    271116 non-null  object
 7   NOC     271116 non-null  object
 8   Games   271116 non-null  object
 9   Year    271116 non-null  int64
 10  Season  271116 non-null  object
 11  City    271116 non-null  object
 12  Sport   271116 non-null  object
 13  Event   271116 non-null  object
 14  Medal   39783 non-null   object
dtypes: float64(3), int64(2), object(10)
memory usage: 31.0+ MB
```

The data is corrected, and it has NA height and NA weight, and after checking it, there was no missing spell. I cleaned the data by dropping data columns that didn't need them and input null height and null weight by average for male and female, null medals to 'none', creating a dummied of medals, sex.

```
In [72]: athlete_events.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 271116 entries, 0 to 271115
Data columns (total 15 columns):
 #   Column  Non-Null Count   Dtype
---  ------  --------------   -----
 0   ID      271116 non-null  int64
 1   Name    271116 non-null  object
 2   Sex     271116 non-null  object
 3   Age     261642 non-null  float64
 4   Height  210945 non-null  float64
 5   Weight  208241 non-null  float64
 6   Team    271116 non-null  object
 7   NOC     271116 non-null  object
 8   Games   271116 non-null  object
 9   Year    271116 non-null  int64
 10  Season  271116 non-null  object
 11  City    271116 non-null  object
 12  Sport   271116 non-null  object
 13  Event   271116 non-null  object
 14  Medal   39783 non-null   object
dtypes: float64(3), int64(2), object(10)
memory usage: 31.0+ MB
```

```
In [73]: avgbysex=athlete_events.groupby('Sex').agg(mean_h=('Height','mean'),mean_w=('Weight','mean'))
         print(avgbysex)
```

```
        mean_h     mean_w
Sex
F   167.839740  60.021252
M   178.858463  75.743677
```

```
In [74]: mean_h_F=avgbysex.loc['F','mean_h']
         mean_h_M=avgbysex.loc['M','mean_h']
         mean_w_F=avgbysex.loc['F','mean_w']
         mean_w_M=avgbysex.loc['M','mean_w']

In [85]: athlete_events.iloc[2,4]

Out[85]: nan

In [75]: athlete_events2=athlete_events.copy()

In [92]: for i in range (0,athlete_events2.shape[0]):
             if athlete_events2.iloc[i,2]=='F' :
                 if math.isnan(athlete_events2.iloc[i,4]) == True :
                     athlete_events2.iloc[i,4]=mean_h_F
                 if math.isnan(athlete_events2.iloc[i,5]) == True :
                     athlete_events2.iloc[i,5]=mean_w_F
             elif  athlete_events2.iloc[i,2]=='M':
                 if math.isnan(athlete_events2.iloc[i,4]) == True :
                     athlete_events2.iloc[i,4]=mean_h_M
                 if math.isnan(athlete_events2.iloc[i,5]) == True :
                     athlete_events2.iloc[i,5]=mean_w_M
```

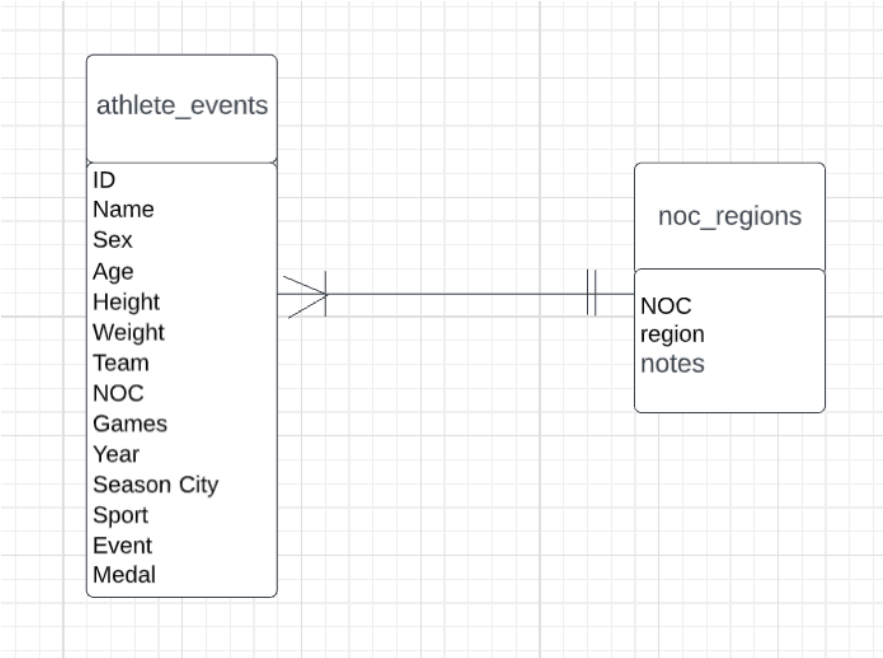
3.Perform initial exploration of data and provide some screenshots or display some stats of the data you are looking at.

In [94]:
```
athlete_events2.describe()
```

Out[94]:

	ID	Age	Height	Weight	Year
count	271116.000000	261642.000000	271116.000000	271116.000000	271116.000000
mean	68248.954396	25.556898	175.829733	71.422034	1978.378480
std	39022.286345	6.393561	9.473486	12.885417	29.877632
min	1.000000	10.000000	127.000000	25.000000	1896.000000
25%	34643.000000	21.000000	170.000000	62.000000	1960.000000
50%	68205.000000	24.000000	178.000000	73.000000	1988.000000
75%	102097.250000	28.000000	180.000000	75.743677	2002.000000
max	135571.000000	97.000000	226.000000	214.000000	2016.000000

4. Create an ERD or proposed ERD to show the relationships of the data you are exploring



Description Write a 5-6 sentence paragraph describing your project; include who might be interested to learn about your findings. Who might be your audience?

Are Age, height, and weight affect the rank of athletes? If we can answer these questions, it will let athletes control their height and weight for better performance.

Questions Create 2-3 questions that you want to answer with the data:

Are Age, height, and weight affect the rank athletes?

Which sports do age, height, and weight affect the rank of athletes?

Hypothesis What are your initial hypotheses about the data?

Age, height, and weight have an effect on some types of sports.

There are differences based on ethnicity.

Approach Describe in 5-6 sentences what approach you are going to take in order to prove (or disprove) your hypotheses.

Using statistics to compare each sport's and ethnicity's age, height, and weight. Creating a bar chart and scatter plot to aid in analysis