In [20]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

Question 1

a.)

In [9]:

```
df = pd.read_csv('HeightWeight.csv')
```

In [12]:

```
df.head()
```

Out[12]:

	Index	Height(Inches)	Weight(Pounds)
0	1	65.78331	112.9925
1	2	71.51521	136.4873
2	3	69.39874	153.0269
3	4	68.21660	142.3354
4	5	67.78781	144.2971

Mean of Height and Weight Columns

In [16]:

```
df.mean()
```

Out[16]:

```
Index              12500.500000
Height(Inches)        67.993114
Weight(Pounds)       127.079421
dtype: float64
```

Standard Deviation of Height and Weight Columns

In [28]:

```
df.std()
```

Out[28]:

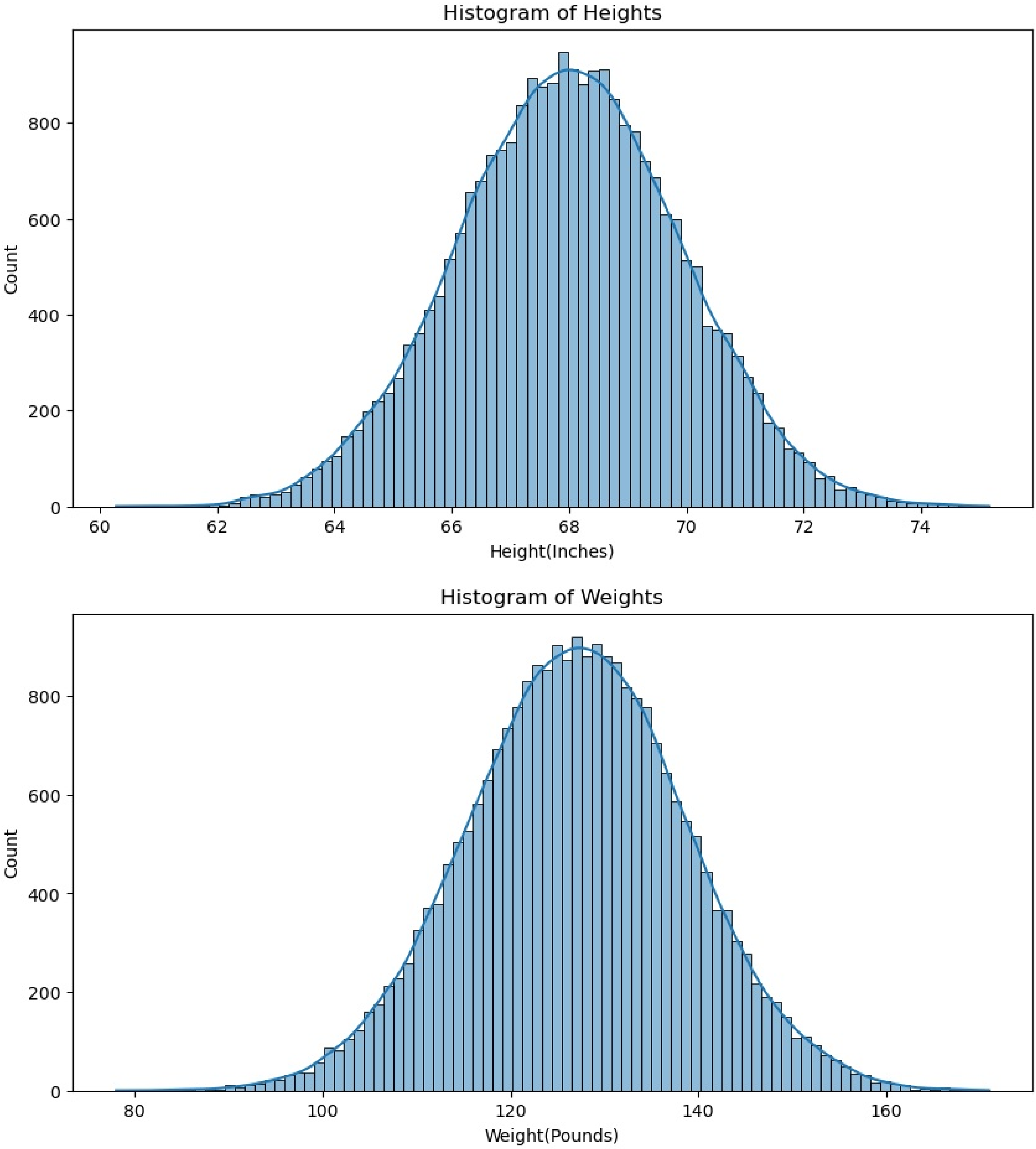
```
Index              7217.022701
Height(Inches)        1.901679
Weight(Pounds)       11.660898
dtype: float64
```

In [30]:

```
plt.figure(figsize=(10, 5))
sns.histplot(x='Height(Inches)', data=df, kde=True)
plt.title('Histogram of Heights')

# Plot histogram for 'Weight(Pounds)'
plt.figure(figsize=(10, 5))
sns.histplot(x='Weight(Pounds)', data=df, kde=True)
plt.title('Histogram of Weights')
plt.show()
```

```
C:\Users\tegbe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is depr
ecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\tegbe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is depr
ecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



Correlation

```
In [41]: df.shape
```

```
Out[41]: (25000, 3)
```

```
In [42]: df_np = df.to_numpy()
         df_np.shape
```

Out[42]:
```
(25000, 3)
```

In [48]:
```
from numpy import corrcoef

cols = 2
df_float = df_np[:,1:3].astype(float)
df_corr = corrcoef(df_float,rowvar=False)

df_corr
```

Out[48]:
```
array([[1.        , 0.50285852],
       [0.50285852, 1.        ]])
```

b.)

In [68]:
```
data = {
    'Time (hours)': [0, 1, 2, 3, 4, 5],
    'Bacteria Count': [100, 180, 324, 583.2, 1049.76, 1889.5]
}

df1 = pd.DataFrame(data)
```

In [94]:
```
from scipy.optimize import curve_fit

#Creating the exponential function
def exponential_func(x, a, b):
    return a * np.exp(b * x)

X = df1['Time (hours)']
y = df1['Bacteria Count']
```

In [95]:
```
#Using the curve_fit module, we can fit into our model and extract the parameters
popt, pcov = curve_fit(exponential_func, X, y)

a, b = popt
print(f"Fitted parameters: a = {a}, b = {b}")
```

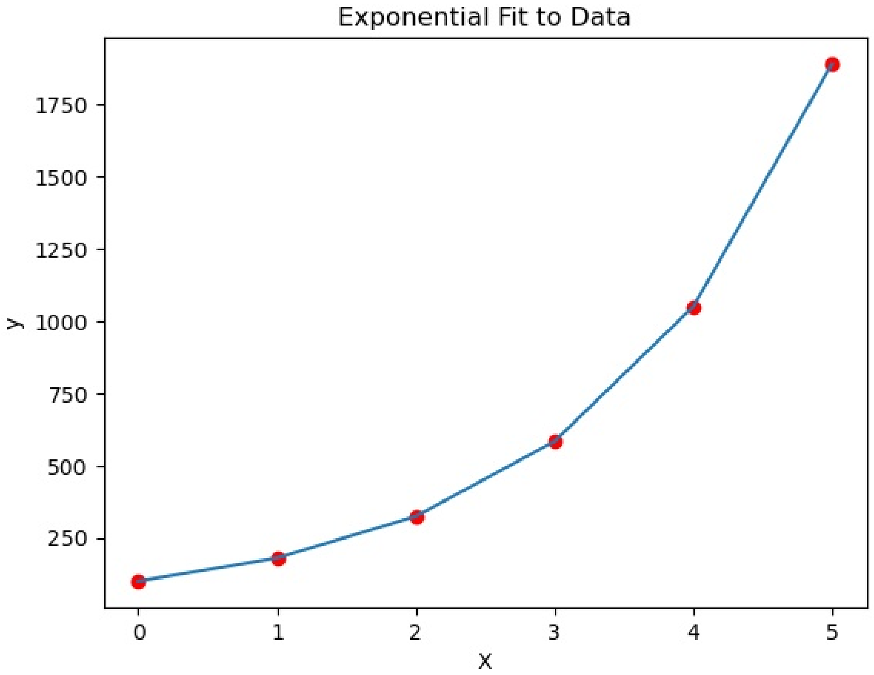
```
Fitted parameters: a = 100.00566045133118, b = 0.5877687861929192
```

In [102…
```
# Plot the original data
plt.plot(X, y, label="Xdata and Ydata")

# Plot the fitted exponential curve
plt.scatter(X, exponential_func(X, *popt), label='Fitted Curve', color='red')

plt.xlabel('X')
plt.ylabel('y')
plt.title('Exponential Fit to Data')
```

Out[102…
```
Text(0.5, 1.0, 'Exponential Fit to Data')
```



R2 Calculation

In [103…

```python
y_pred = exponential_func(X, *popt)
```

Out[103…

```
0     100.005660
1     180.006970
2     324.006754
3     583.201730
4    1049.744346
5    1889.506041
Name: Time (hours), dtype: float64
```

In [113…

```python
y_pred = exponential_func(X, *popt)

# Calculate R^2
SS_res = np.sum((y - y_pred) ** 2)  # Residual sum of squares
SS_tot = np.sum((y - np.mean(y)) ** 2)  # Total sum of squares
R2 = 1 - (SS_res / SS_tot)

print(f"Coefficient (a): {popt[0]}")
print(f"Growth Rate (b): {popt[1]}")
print(f"Coefficient of Determination (R²): {R2}")
```

```
Coefficient (a): 100.00566045133118
Growth Rate (b): 0.5877687861929192
Coefficient of Determination (R²): 0.9999999998230766
```

In [ ]:

Question 2

a.) TASK 1

In [114…

```python
df2 = pd.read_csv('task1.txt')
df2.head()
```

Out[114…

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	CRSArrTime	UniqueCarrier	FlightNum	...	TaxiIn	Ta
0	2008	1	1	2	120.0	1935	309.0	2130	9E	5746	...	3.0	
1	2008	1	1	2	555.0	600	826.0	835	AA	1614	...	7.0	
2	2008	1	1	2	600.0	600	728.0	729	YV	2883	...	7.0	
3	2008	1	1	2	601.0	605	727.0	750	9E	5743	...	4.0	
4	2008	1	1	2	601.0	600	654.0	700	AA	1157	...	5.0	

5 rows × 29 columns

In [115…

```python
df2.columns
```

Out[115…

```
Index(['Year', 'Month', 'DayofMonth', 'DayOfWeek', 'DepTime', 'CRSDepTime',
       'ArrTime', 'CRSArrTime', 'UniqueCarrier', 'FlightNum', 'TailNum',
       'ActualElapsedTime', 'CRSElapsedTime', 'AirTime', 'ArrDelay',
       'DepDelay', 'Origin', 'Dest', 'Distance', 'TaxiIn', 'TaxiOut',
       'Cancelled', 'CancellationCode', 'Diverted', 'CarrierDelay',
       'WeatherDelay', 'NASDelay', 'SecurityDelay', 'LateAircraftDelay'],
      dtype='object')
```

What time of the day it is best to fly so as to have the least possible delays?

In [162…

```python
new_df = df2[['Month','DepTime','ArrDelay',
       'DepDelay', 'Origin', 'Dest', 'Distance']]

new_df
```

Out[162…

	Month	DepTime	ArrDelay	DepDelay	Origin	Dest	Distance
0	1	120.0	339.0	345.0	MEM	AUS	559
1	1	555.0	-9.0	-5.0	AUS	ORD	978
2	1	600.0	-1.0	0.0	AUS	PHX	872
3	1	601.0	-23.0	-4.0	AUS	MEM	559
4	1	601.0	-6.0	1.0	AUS	DFW	190
...	...	...	...	...	...	...	...
99255	12	2109.0	1.0	9.0	IAH	AUS	140
99256	12	2124.0	-14.0	1.0	ATL	AUS	813
99257	12	2327.0	-11.0	2.0	DFW	AUS	190
99258	12	NaN	NaN	NaN	AUS	BOS	1698
99259	12	NaN	NaN	NaN	JFK	AUS	1522

99260 rows × 7 columns

We would then drop all NaN values/rows

In [143…

```
new_df.dropna(axis=0,inplace=True)
```

```
C:\Users\tegbe\AppData\Local\Temp\ipykernel_13988\4151371568.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#retu
rning-a-view-versus-a-copy
  new_df.dropna(axis=0,inplace=True)
```

In [163…

```
new_df
```

Out[163…

	Month	DepTime	ArrDelay	DepDelay	Origin	Dest	Distance
0	1	120.0	339.0	345.0	MEM	AUS	559
1	1	555.0	-9.0	-5.0	AUS	ORD	978
2	1	600.0	-1.0	0.0	AUS	PHX	872
3	1	601.0	-23.0	-4.0	AUS	MEM	559
4	1	601.0	-6.0	1.0	AUS	DFW	190
...	...	...	...	...	...	...	...
99255	12	2109.0	1.0	9.0	IAH	AUS	140
99256	12	2124.0	-14.0	1.0	ATL	AUS	813
99257	12	2327.0	-11.0	2.0	DFW	AUS	190
99258	12	NaN	NaN	NaN	AUS	BOS	1698
99259	12	NaN	NaN	NaN	JFK	AUS	1522

99260 rows × 7 columns

In [179…

```
high_delays_g = new_df.groupby('DepTime')['DepDelay'].sum().reset_index()
high_delays_g
```

Out[179…

	DepTime	DepDelay
0	1.0	683.0
1	2.0	129.0
2	3.0	249.0
3	4.0	1029.0
4	5.0	155.0
...	...	...
1203	2356.0	76.0
1204	2357.0	547.0
1205	2358.0	744.0
1206	2359.0	74.0
1207	2400.0	360.0

1208 rows × 2 columns

In [228…

```
high_delays = high_delays_g[high_delays_g['DepDelay']<-405]
high_delays.sort_values('DepDelay',ascending=True,inplace=True)
high_delays
```

```
C:\Users\tegbe\AppData\Local\Temp\ipykernel_13988\1390945243.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#retu
rning-a-view-versus-a-copy
  high_delays.sort_values('DepDelay',ascending=True,inplace=True)
```

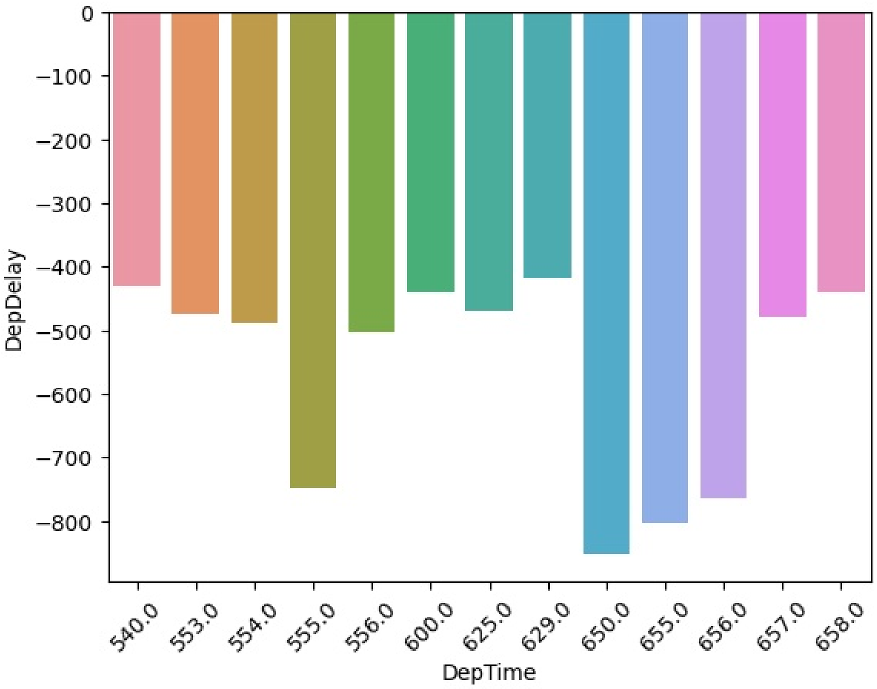
Out[228…

	DepTime	DepDelay
177	650.0	-852.0
182	655.0	-804.0
183	656.0	-766.0
122	555.0	-747.0
123	556.0	-504.0
121	554.0	-490.0
184	657.0	-480.0
120	553.0	-475.0
152	625.0	-470.0
127	600.0	-441.0
185	658.0	-441.0
107	540.0	-430.0
156	629.0	-419.0

The time with the -852 DepDelay is the best time to fly which is 650.0 Departure Time

In [231…

```
sns.barplot(x=high_delays['DepTime'],y=high_delays['DepDelay'])
plt.xticks(rotation=45)
plt.show()
```



In [ ]:

```

```

What time of the year it is more suited to fly so as to have the delays minimum and does the destination affect this?

In [232…

```
new_df
```

Out[232…

	Month	DepTime	ArrDelay	DepDelay	Origin	Dest	Distance
0	1	120.0	339.0	345.0	MEM	AUS	559
1	1	555.0	-9.0	-5.0	AUS	ORD	978
2	1	600.0	-1.0	0.0	AUS	PHX	872
3	1	601.0	-23.0	-4.0	AUS	MEM	559
4	1	601.0	-6.0	1.0	AUS	DFW	190
...	...	...	...	...	...	...	...
99255	12	2109.0	1.0	9.0	IAH	AUS	140
99256	12	2124.0	-14.0	1.0	ATL	AUS	813
99257	12	2327.0	-11.0	2.0	DFW	AUS	190
99258	12	NaN	NaN	NaN	AUS	BOS	1698
99259	12	NaN	NaN	NaN	JFK	AUS	1522

99260 rows × 7 columns

In [294…

```
y_delays = new_df.groupby('Month')['DepDelay'].mean().reset_index()
y_delays
```

Out[294…

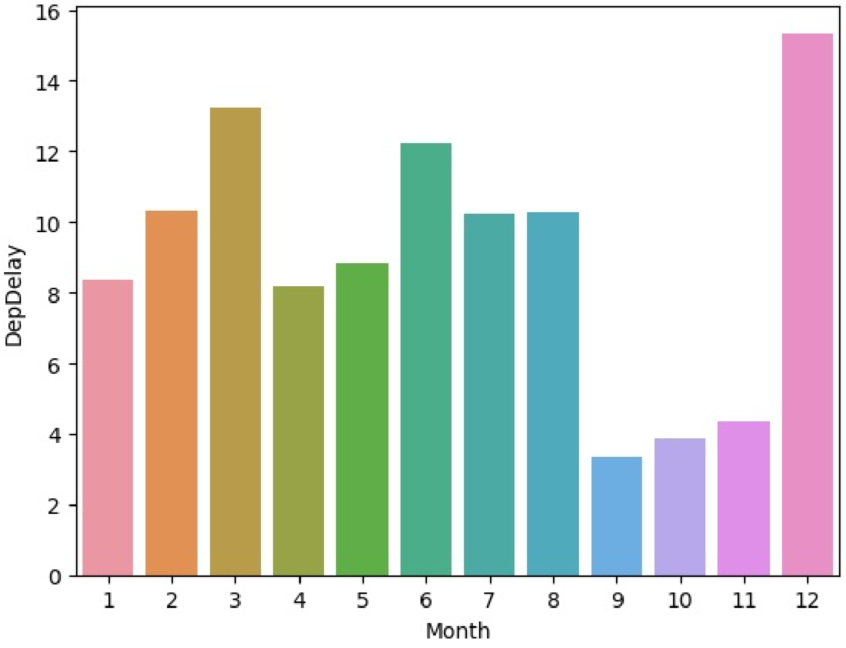
	Month	DepDelay
0	1	8.371216
1	2	10.308328
2	3	13.253937
3	4	8.174785
4	5	8.852347
5	6	12.232877
6	7	10.231535
7	8	10.257626
8	9	3.327223
9	10	3.877845
10	11	4.357531
11	12	15.337521

In [295…

```
sns.barplot(x=y_delays['Month'],y=y_delays['DepDelay'])
```

Out[295…

```
<Axes: xlabel='Month', ylabel='DepDelay'>
```



September is the best time of the month as it has the less number of delays

In [247…

```
dest = new_df.groupby(['Month', 'Dest'])['DepDelay'].sum().reset_index()
dest
```

Out[247…

	Month	Dest	DepDelay
0	1	ABQ	389.0
1	1	ATL	1650.0
2	1	AUS	45299.0
3	1	BNA	357.0
4	1	BOS	186.0
...	...	...	...
502	12	SAN	596.0
503	12	SFO	1141.0
504	12	SJC	140.0
505	12	SLC	125.0
506	12	TPA	776.0

507 rows × 3 columns

In [260…

```
### Getting the destinations with less delays in September
```

```
sept = dest[dest['Month'] == 9]
sept['Dest'].value_counts()
```

Out[260…

```
Dest
ABQ    1
ORD    1
LBB    1
LGB    1
MAF    1
MCO    1
MDW    1
MEM    1
MSY    1
OAK    1
PHL    1
ATL    1
PHX    1
RDU    1
SAN    1
SEA    1
SFO    1
SJC    1
SLC    1
SNA    1
LAX    1
LAS    1
JFK    1
IND    1
AUS    1
BNA    1
BOS    1
BWI    1
CLE    1
CLT    1
CVG    1
DAL    1
DEN    1
DFW    1
ELP    1
EWR    1
FLL    1
HOU    1
HRL    1
IAD    1
IAH    1
TPA    1
Name: count, dtype: int64
```

In [255…

```
# Getting the destinations with less delays throughout the whole year

dest_g = dest[dest['DepDelay']<-10]
dest_g['Dest'].value_counts()
```

Out[255…

```
Dest
SNA    5
IND    4
SLC    4
MEM    4
CLT    4
SJC    4
DFW    3
TUS    3
SFO    3
BOS    2
MCI    2
JAX    2
TUL    2
STL    2
MSY    1
LGB    1
CVG    1
MDW    1
MCO    1
IAH    1
FLL    1
DEN    1
CLE    1
ATL    1
SEA    1
MAF    1
MSP    1
ABQ    1
IAD    1
Name: count, dtype: int64
```

As we can see from above SNA, IND, SLC, MEM are the destinations with less delays however they are not so present in September

Therefore the destination doesn't affect the time of the year with minimum delays

In [ ]:

```

```

In [ ]:

```

```

Investigate on how the pattern of flights to various destinations alter over the course of year

In [287…

```
month = new_df.groupby(['Month', 'Dest']).size().reset_index(name='Count')
month
```

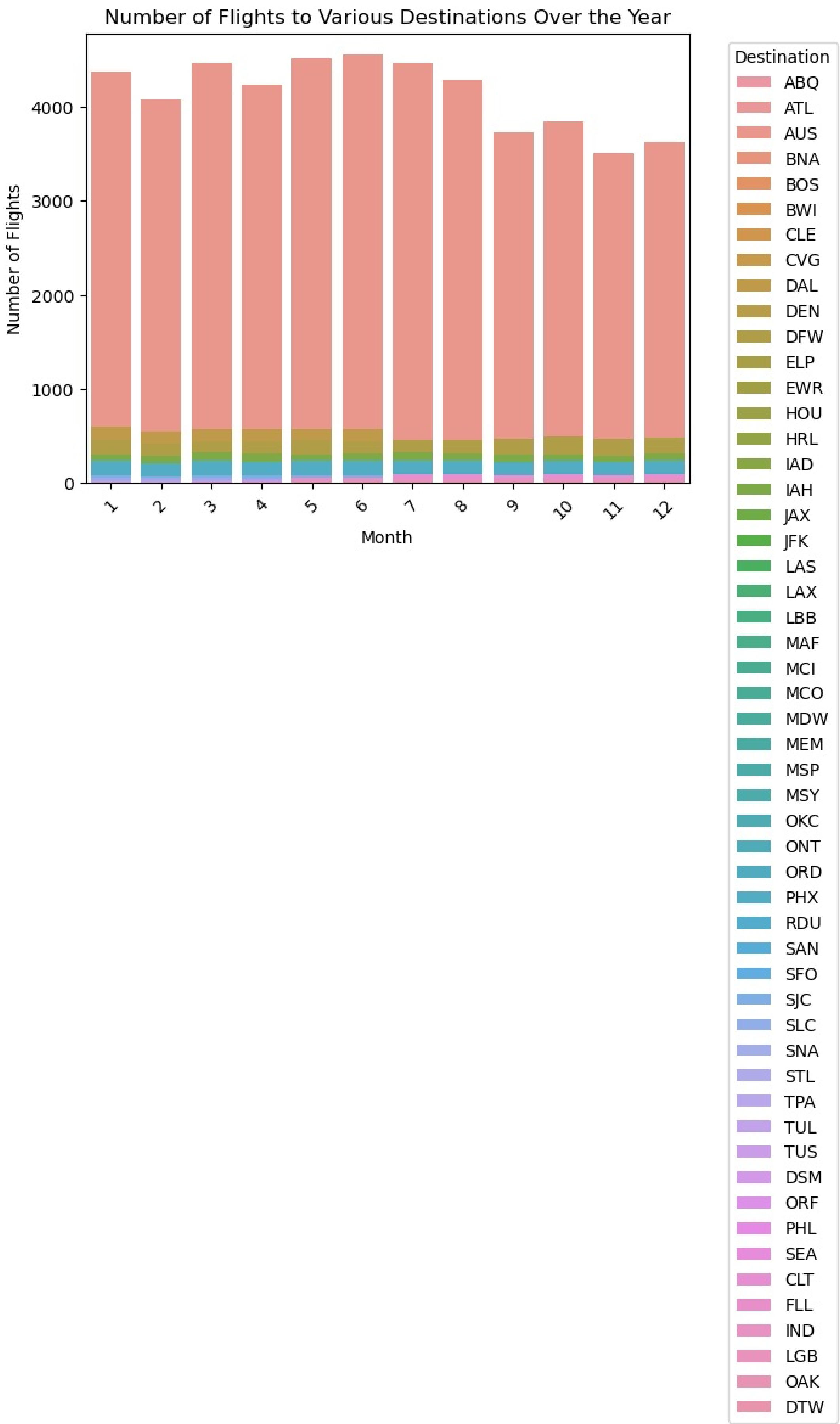
Out[287…

	Month	Dest	Count
0	1	ABQ	58
1	1	ATL	210
2	1	AUS	4365
3	1	BNA	82
4	1	BOS	31
...	...	...	...
502	12	SAN	62
503	12	SFO	62
504	12	SJC	77
505	12	SLC	31
506	12	TPA	31

507 rows × 3 columns

In [307…

```
sns.barplot(x=month['Month'], y=month['Count'],hue=month['Dest'],dodge=False)
plt.title('Number of Flights to Various Destinations Over the Year')
plt.ylabel('Number of Flights')
plt.xlabel('Month')
plt.xticks(rotation=45)
plt.legend(title='Destination', bbox_to_anchor=(1.05, 1), loc='upper left')
plt.show()
```



In [ ]:

In [ ]:

b.) TASK 2

In [308…:

```
df3 = pd.read_csv('task2.txt')
df3.head()
```

Out[308…

	id	name	sex	age	height	weight	team	noc	games	year	season	city	sport	event	medal
0	16	Juhamatti Tapio Aaltonen	M	28	184	85.0	Finland	FIN	2014 Winter	2014	Winter	Sochi	Ice Hockey	Ice Hockey Men's Ice Hockey	Bronze
1	17	Paavo Johannes Aaltonen	M	28	175	64.0	Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Individual All-Around	Bronze
2	17	Paavo Johannes Aaltonen	M	28	175	64.0	Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Team All-Around	Gold
3	17	Paavo Johannes Aaltonen	M	28	175	64.0	Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Horse Vault	Gold
4	17	Paavo Johannes Aaltonen	M	28	175	64.0	Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Pommelled Horse	Gold

QUESTION: Compute the 95th percentile of heights for the competitors in all Athletic events for gender Female. Note that sport refers to the broad sports (Athletics) and event is the specific event (100-meter sprint)

In [310…

```
df3['sport'].unique()
```

Out[310…

```
array(['Ice Hockey', 'Gymnastics', 'Rowing', 'Football', 'Fencing',
       'Athletics', 'Canoeing', 'Handball', 'Water Polo', 'Wrestling',
       'Sailing', 'Cycling', 'Hockey', 'Swimming', 'Boxing', 'Basketball',
       'Volleyball', 'Cross Country Skiing', 'Equestrianism', 'Shooting'],
      dtype=object)
```

In [311…

```
df3['event'].unique()
```

Out[311…

```
array(["Ice Hockey Men's Ice Hockey",
       "Gymnastics Men's Individual All-Around",
       "Gymnastics Men's Team All-Around", "Gymnastics Men's Horse Vault",
       "Gymnastics Men's Pommelled Horse",
       "Rowing Men's Lightweight Double Sculls",
       "Football Women's Football", "Rowing Men's Coxless Pairs",
       "Fencing Men's epee, Individual",
       "Athletics Women's Javelin Throw",
       "Canoeing Men's Kayak Fours, 1,000 metres",
       "Handball Men's Handball", "Football Men's Football",
       "Water Polo Men's Water Polo",
       "Wrestling Men's Featherweight, Freestyle",
       'Sailing Mixed Two Person Heavyweight Dinghy',
       "Athletics Men's 1,500 metres", "Cycling Women's Sprint",
       "Fencing Men's Sabre, Team", "Rowing Men's Quadruple Sculls",
       "Rowing Men's Double Sculls", "Rowing Men's Coxed Pairs",
       "Hockey Men's Hockey", "Wrestling Men's Middleweight, Greco-Roman",
       "Swimming Women's 4 x 100 metres Medley Relay",
       "Boxing Men's Light-Welterweight",
       "Wrestling Men's Flyweight, Freestyle",
       "Wrestling Men's Bantamweight, Freestyle",
       "Basketball Men's Basketball",
       "Wrestling Men's Light-Heavyweight, Freestyle",
       "Athletics Men's Hammer Throw", "Athletics Men's 10,000 metres",
       "Gymnastics Women's Team All-Around", "Athletics Men's Marathon",
       "Athletics Men's Long Jump", "Athletics Men's Triple Jump",
       "Athletics Women's 5,000 metres",
       "Wrestling Men's Middleweight, Freestyle",
       "Athletics Men's 4 x 400 metres Relay",
       "Gymnastics Men's Floor Exercise", "Gymnastics Men's Rings",
       "Swimming Men's 4 x 100 metres Freestyle Relay",
       "Swimming Men's 4 x 200 metres Freestyle Relay",
       "Fencing Men's Foil, Individual",
       "Canoeing Men's Kayak Doubles, 500 metres",
       "Swimming Men's 4 x 100 metres Medley Relay",
       "Wrestling Men's Light-Heavyweight, Greco-Roman",
       "Athletics Men's 100 metres",
       "Athletics Men's 4 x 100 metres Relay",
       "Volleyball Men's Volleyball", "Volleyball Women's Volleyball",
       "Basketball Women's Basketball",
       "Wrestling Men's Lightweight, Freestyle",
       "Cycling Men's Mountainbike, Cross-Country",
       "Fencing Men's epee, Team", "Boxing Men's Welterweight",
       "Wrestling Men's Flyweight, Greco-Roman",
       "Cycling Men's 100 kilometres Team Time Trial",
       "Boxing Men's Featherweight", "Boxing Men's Bantamweight",
       "Cross Country Skiing Women's 3 x 5 kilometres Relay",
       "Rowing Women's Coxed Eights", "Rowing Women's Quadruple Sculls",
       "Boxing Men's Heavyweight", "Boxing Men's Lightweight",
       "Athletics Men's Standing High Jump",
       "Athletics Men's Standing Long Jump", "Boxing Women's Flyweight",
       "Athletics Women's Shot Put", "Rowing Men's Coxed Eights",
       "Canoeing Men's Canadian Doubles, Slalom",
       "Athletics Men's 400 metres Hurdles",
       'Sailing Mixed Three Person Keelboat',
       "Swimming Women's 400 metres Freestyle",
       "Swimming Women's 800 metres Freestyle",
       "Swimming Men's 100 metres Freestyle",
       "Swimming Men's 50 metres Freestyle",
       "Athletics Women's 4 x 400 metres Relay",
       "Rowing Women's Coxed Quadruple Sculls",
       "Gymnastics Women's Horse Vault", "Rowing Men's Coxless Fours",
       "Hockey Women's Hockey", "Swimming Men's 200 metres Freestyle",
       "Ice Hockey Women's Ice Hockey", "Fencing Women's Foil, Team",
       "Handball Women's Handball", 'Equestrianism Mixed Jumping, Team',
       "Swimming Women's 4 x 100 metres Freestyle Relay",
       "Swimming Women's 100 metres Butterfly",
       "Water Polo Women's Water Polo",
       "Canoeing Men's Kayak Singles, Slalom",
       'Sailing Mixed One Person Dinghy',
       "Sailing Men's One Person Dinghy",
       "Sailing Men's One Person Heavyweight Dinghy",
       'Sailing Mixed 5.5 metres',
       "Cycling Men's Team Pursuit, 4,000 metres",
       "Cycling Men's Madison",
       "Shooting Men's Small-Bore Rifle, Prone, 50 metres",
       "Athletics Women's Long Jump",
       "Wrestling Men's Super-Heavyweight, Freestyle",
       "Fencing Men's Foil, Team",
       "Wrestling Men's Heavyweight, Freestyle",
       "Athletics Women's 4 x 100 metres Relay",
       "Fencing Women's Sabre, Team", 'Sailing Mixed Two Person Keelboat',
       "Shooting Men's Skeet", "Shooting Men's Double Trap",
       "Shooting Men's Trap", 'Equestrianism Mixed Jumping, Individual',
       "Sailing Women's Windsurfer",
       "Cross Country Skiing Men's 4 x 10 kilometres Relay",
       "Cross Country Skiing Men's 10 kilometres",
       "Wrestling Men's Lightweight, Greco-Roman",
       "Cross Country Skiing Women's 4 x 5 kilometres Relay",
       "Athletics Men's High Jump", "Sailing Women's Two Person Dinghy",
       "Athletics Men's Discus Throw", "Rowing Women's Double Sculls",
       "Wrestling Men's Heavyweight, Greco-Roman",
       "Wrestling Men's Light-Flyweight, Greco-Roman",
       "Shooting Men's Air Rifle, 10 metres",
       'Equestrianism Mixed Three-Day Event, Individual',
       "Athletics Women's 100 metres Hurdles",
       "Shooting Men's Rapid-Fire Pistol, 25 metres",
       "Shooting Men's Small-Bore Rifle, Three Positions, 50 metres",
       'Equestrianism Mixed Three-Day Event, Team',
       "Athletics Men's Cross-Country, Team", "Boxing Men's Flyweight",
       "Cross Country Skiing Men's 30 kilometres",
       "Cross Country Skiing Men's 10/15 kilometres Pursuit",
       "Cross Country Skiing Men's 10/10 kilometres Pursuit",
       "Swimming Women's 50 metres Freestyle",
       "Swimming Women's 100 metres Freestyle",
       "Rowing Women's Coxless Pairs",
       "Athletics Men's 10 kilometres Walk",
       "Boxing Men's Super-Heavyweight",
       "Cross Country Skiing Men's Team Sprint",
       "Gymnastics Women's Individual All-Around",
       "Gymnastics Women's Floor Exercise",
       "Rowing Men's Lightweight Coxless Fours",
       "Boxing Men's Middleweight", "Sailing Women's One Person Dinghy",
       "Athletics Women's 400 metres", "Athletics Men's 800 metres",
       "Wrestling Men's Featherweight, Greco-Roman",
       "Wrestling Women's Lightweight, Freestyle",
       "Canoeing Men's Canadian Singles, 1,000 metres",
       "Wrestling Men's Bantamweight, Greco-Roman",
       "Canoeing Men's Canadian Doubles, 500 metres",
       "Athletics Men's 3,000 metres Steeplechase",
       "Athletics Women's Marathon",
       "Swimming Women's 4 x 200 metres Freestyle Relay",
       "Shooting Men's Free Rifle, Three Positions, 300 metres",
       'Shooting Mixed Free Rifle, Three Positions, 300 metres',
       "Cycling Men's Individual Pursuit, 4,000 metres",
       "Swimming Women's 10 kilometres Open Water",
       'Sailing Mixed Multihull', "Athletics Men's 110 metres Hurdles",
       "Canoeing Women's Kayak Singles, 500 metres",
       "Canoeing Women's Kayak Doubles, 500 metres",
       "Canoeing Women's Kayak Fours, 500 metres",
       "Athletics Men's Cross-Country, Individual",
           "Cross Country Skiing Women's Team Sprint",
           "Athletics Men's Shot Put",
           "Swimming Women's 100 metres Backstroke",
           "Athletics Men's 20 kilometres Walk",
           "Gymnastics Men's Parallel Bars",
           "Gymnastics Men's Horizontal Bar",
           "Cross Country Skiing Men's 15 kilometres",
           "Cross Country Skiing Men's 30 km Skiathlon",
           "Rowing Women's Coxed Fours",
           "Swimming Women's 100 metres Breaststroke",
           "Canoeing Men's Kayak Singles, 1,000 metres",
           "Athletics Women's Discus Throw", "Rowing Women's Single Sculls",
           "Athletics Women's 400 metres Hurdles",
           "Athletics Men's 5,000 metres", "Athletics Men's 200 metres",
           "Fencing Men's Sabre, Individual",
           "Athletics Men's 50 kilometres Walk", "Boxing Women's Lightweight",
           "Athletics Men's Pole Vault", "Cycling Women's Team Pursuit",
           "Boxing Men's Light-Flyweight", "Boxing Men's Light-Heavyweight",
           "Cycling Women's Road Race, Individual",
           "Swimming Men's 400 metres Freestyle",
           "Cycling Women's Individual Time Trial",
           "Cycling Women's Individual Pursuit, 3,000 metres",
           "Wrestling Men's Welterweight, Greco-Roman",
           "Swimming Men's 100 metres Butterfly",
           "Athletics Women's 100 metres", "Sailing Men's Windsurfer",
           "Athletics Women's 3,000 metres Steeplechase",
           "Athletics Women's High Jump",
           "Gymnastics Women's Team Portable Apparatus",
           "Gymnastics Women's Uneven Bars",
           "Shooting Women's Sporting Pistol, 25 metres",
           "Wrestling Men's Welterweight, Freestyle",
           "Swimming Women's 200 metres Backstroke",
           "Rowing Men's Coxed Fours",
           'Shooting Mixed Small-Bore Rifle, Prone, 50 metres',
           "Cross Country Skiing Women's 10 kilometres",
           "Athletics Men's Decathlon",
           "Cross Country Skiing Women's 5 kilometres",
           "Shooting Men's Running Target, 50 metres",
           "Canoeing Men's Canadian Singles, Slalom", "Cycling Men's Keirin",
           "Canoeing Men's Kayak Doubles, 1,000 metres",
           "Canoeing Men's Kayak Doubles, 10,000 metres",
           "Athletics Women's 10,000 metres",
           "Sailing Women's Three Person Keelboat",
           "Fencing Women's epee, Team",
           "Swimming Women's 200 metres Breaststroke",
           "Swimming Women's 200 metres Freestyle",
           "Athletics Men's 400 metres",
           "Shooting Men's Free Pistol, 50 metres",
           "Shooting Men's Air Pistol, 10 metres",
           "Swimming Men's 200 metres Butterfly",
           "Athletics Men's Javelin Throw",
           "Athletics Men's 3,000 metres, Team", "Shooting Women's Skeet",
           "Canoeing Men's Kayak Singles, 500 metres",
           "Canoeing Women's Kayak Singles, Slalom",
           "Athletics Women's 1,500 metres",
           "Swimming Men's 200 metres Breaststroke", 'Shooting Mixed Trap',
           "Boxing Men's Light-Middleweight",
           'Equestrianism Mixed Dressage, Individual',
           'Equestrianism Mixed Dressage, Team',
           "Cycling Women's 500 metres Time Trial", "Shooting Women's Trap",
           "Swimming Men's 200 metres Backstroke",
           "Swimming Women's 400 metres Individual Medley",
           "Cycling Men's Road Race, Individual", 'Sailing Mixed Skiff',
           "Fencing Women's epee, Individual", "Rowing Women's Coxless Fours",
           "Swimming Men's 100 metres Backstroke",
           "Wrestling Men's Super-Heavyweight, Greco-Roman",
           "Sailing Men's Two Person Dinghy",
           "Gymnastics Men's Rope Climbing",
           'Shooting Mixed Small-Bore Rifle, Three Positions, 50 metres',
           "Shooting Women's Air Pistol, 10 metres",
           "Swimming Men's 400 metres Individual Medley",
           "Fencing Women's Foil, Individual",
           "Cross Country Skiing Women's 5/5 kilometres Pursuit",
           "Cycling Men's Team Sprint", "Cycling Men's Sprint",
           "Swimming Men's 200 metres Individual Medley",
           "Swimming Women's 200 metres Individual Medley",
           "Canoeing Men's Kayak Singles, 200 metres",
           "Swimming Men's 1,500 metres Freestyle",
           "Cycling Men's Tandem Sprint, 2,000 metres",
           "Athletics Women's Heptathlon",
           "Rowing Women's Lightweight Double Sculls",
           'Shooting Mixed Running Target, 50 metres',
           "Cycling Women's Points Race",
           "Cross Country Skiing Women's 30 kilometres",
           "Cross Country Skiing Women's 5/10 kilometres Pursuit",
           "Cross Country Skiing Women's 15 kilometres",
           "Swimming Women's 200 metres Butterfly",
           "Athletics Women's 800 metres", 'Sailing Mixed Two Person Dinghy',
           "Shooting Women's Small-Bore Rifle, Three Positions, 50 metres",
           "Athletics Men's Pentathlon", "Athletics Women's Pentathlon",
           "Gymnastics Women's Balance Beam",
           "Shooting Women's Air Rifle, 10 metres",
           "Cross Country Skiing Women's Sprint",
           "Cross Country Skiing Women's 15 km Skiathlon",
           "Athletics Women's 200 metres", "Cycling Men's Road Race, Team",
           "Athletics Women's 80 metres Hurdles",
           "Cycling Women's Mountainbike, Cross-Country",
           "Shooting Men's Running Target, 10 metres",
           "Cycling Men's Individual Time Trial",
           "Canoeing Men's Canadian Doubles, 1,000 metres",
           "Athletics Men's 1,500 metres Walk",
           "Athletics Men's 3 mile, Team",
           "Cross Country Skiing Men's 50 kilometres",
           "Athletics Men's 1,600 metres Medley Relay",
           "Rowing Men's Single Sculls",
           "Swimming Men's 100 metres Breaststroke", 'Shooting Mixed Skeet',
           "Equestrianism Men's Three-Day Event, Individual",
           "Equestrianism Men's Three-Day Event, Team",
           "Canoeing Men's Canadian Singles, 500 metres",
           "Wrestling Women's Heavyweight, Freestyle", "Sailing Men's Skiff",
           "Canoeing Women's Kayak Singles, 200 metres",
           "Cycling Men's Omnium", "Athletics Women's 3,000 metres",
           "Cycling Women's BMX",
           "Canoeing Men's Canadian Singles, 200 metres",
           "Athletics Women's 10 kilometres Walk",
           "Wrestling Women's Flyweight, Freestyle",
           "Cycling Men's 1,000 metres Time Trial",
           "Canoeing Men's Kayak Doubles, 200 metres",
           "Athletics Women's Hammer Throw",
           "Athletics Men's 2,590 metres Steeplechase",
           "Swimming Men's 50 yard Freestyle",
           "Swimming Men's 100 yard Freestyle",
           "Swimming Men's 220 yard Freestyle",
           "Swimming Men's 440 yard Freestyle",
           "Swimming Men's 4 x 50 Yard Freestyle Relay",
           "Cross Country Skiing Men's Sprint",
           "Swimming Men's 10 kilometres Open Water", "Cycling Men's BMX",
           "Athletics Men's 56-pound Weight Throw",
           "Athletics Women's Triple Jump", "Cycling Women's Omnium",
           "Wrestling Men's Light-Flyweight, Freestyle",
           'Shooting Mixed Free Pistol, 50 metres',
           "Athletics Men's Stone Throw",
           "Athletics Men's Javelin Throw, Freestyle",
           "Wrestling Women's Light-Heavyweight, Freestyle",
           "Athletics Women's Pole Vault",
           "Canoeing Men's Folding Kayak Singles, 10 kilometres",
           "Athletics Men's 3,200 metres Steeplechase",
           "Sailing Men's Two Person Keelboat",
           "Athletics Men's Standing Triple Jump",
           'Shooting Mixed Rapid-Fire Pistol, 25 metres',
           "Canoeing Men's Canadian Doubles, 10,000 metres",
           'Sailing Mixed 6 metres', "Boxing Women's Middleweight",
           "Canoeing Men's Kayak Singles, 10,000 metres",
           "Athletics Men's 3,000 metres Walk", "Cycling Men's Points Race",
           "Shooting Women's Double Trap", 'Sailing Mixed Windsurfer',
           "Athletics Men's Discus Throw, Greek Style",
           "Cycling Women's Team Sprint",
           "Athletics Women's 20 kilometres Walk", "Sailing Women's Skiff",
           "Cycling Women's Keirin", "Equestrianism Men's Jumping, Team",
           "Athletics Men's 60 metres",
           "Canoeing Men's Kayak Relay 4 x 500 metres",
           "Athletics Men's All-Around Championship",
           "Athletics Men's 4 mile, Team",
           "Canoeing Men's Canadian Singles, 10,000 metres",
           "Athletics Men's 200 metres Hurdles", "Shooting Men's Trap, Team",
           "Wrestling Women's Middleweight, Freestyle",
           "Fencing Women's Sabre, Individual",
           "Cross Country Skiing Women's 20 kilometres",
           "Equestrianism Men's Jumping, Individual",
           "Athletics Men's 3,500 metres Walk",
           "Wrestling Men's Middleweight A, Greco-Roman",
           "Athletics Men's 10 mile Walk",
           "Cross Country Skiing Men's 18 kilometres",
           "Athletics Men's Pentathlon (Ancient)",
           "Athletics Men's Discus Throw, Both Hands",
           "Wrestling Women's Featherweight, Freestyle",
              "Athletics Men's Shot Put, Both Hands", 'Sailing Mixed 6.5 metres',
              "Athletics Men's 2,500 metres Steeplechase",
              "Athletics Men's 5 mile",
              "Athletics Men's Javelin Throw, Both Hands",
              "Athletics Men's 5,000 metres, Team", "Cycling Men's 5,000 metres",
              "Gymnastics Men's Parallel Bars, Teams",
              "Wrestling Men's Unlimited Class, Greco-Roman",
              "Gymnastics Men's Horizontal Bar, Teams",
              "Gymnastics Men's Team All-Around, Swedish System",
              "Equestrianism Men's Dressage, Team",
              "Swimming Men's One Mile Freestyle",
              "Swimming Men's 4 x 250 metres Freestyle Relay",
              "Swimming Men's 1,000 metres Freestyle",
              "Swimming Men's 4,000 metres Freestyle",
              "Wrestling Men's All-Around, Greco-Roman"], dtype=object)
```

In [317…

```
df3[df3['event'].str.contains('100 sprint')]
```

Out[317…

id	name	sex	age	height	weight	team	noc	games	year	season	city	sport	event	medal

In [322…

```
# Derving the data whose sex is Female
fdata = df3[(df3['sport'] == 'Athletics') & (df3['sex'] == 'F')]
fdata
```

Out[322…

	id	name	sex	age	height	weight	team	noc	games	year	season	city	sport	event	med
10	67	Mariya Vasilyevna Abakumova (-Tarabina)	F	22	179	80.0	Russia	RUS	2008 Summer	2008	Summer	Beijing	Athletics	Athletics Women's Javelin Throw	Silv
50	428	Elvan Abeylegesse	F	25	159	40.0	Turkey	TUR	2008 Summer	2008	Summer	Beijing	Athletics	Athletics Women's 5,000 metres	Silv
103	846	Valerie Kasanita Adams-Vili (-Price)	F	23	193	120.0	New Zealand	NZL	2008 Summer	2008	Summer	Beijing	Athletics	Athletics Women's Shot Put	Go
104	846	Valerie Kasanita Adams-Vili (-Price)	F	27	193	120.0	New Zealand	NZL	2012 Summer	2012	Summer	London	Athletics	Athletics Women's Shot Put	Go
105	846	Valerie Kasanita Adams-Vili (-Price)	F	31	193	120.0	New Zealand	NZL	2016 Summer	2016	Summer	Rio de Janeiro	Athletics	Athletics Women's Shot Put	Silv
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
23844	135553	Galina Ivanovna Zybina (-Fyodorova)	F	21	168	80.0	Soviet Union	URS	1952 Summer	1952	Summer	Helsinki	Athletics	Athletics Women's Shot Put	Go
23845	135553	Galina Ivanovna Zybina (-Fyodorova)	F	25	168	80.0	Soviet Union	URS	1956 Summer	1956	Summer	Melbourne	Athletics	Athletics Women's Shot Put	Silv
23846	135553	Galina Ivanovna Zybina (-Fyodorova)	F	33	168	80.0	Soviet Union	URS	1964 Summer	1964	Summer	Tokyo	Athletics	Athletics Women's Shot Put	Bronz
23848	135563	Olesya Nikolayevna Zykina	F	19	171	64.0	Russia	RUS	2000 Summer	2000	Summer	Sydney	Athletics	Athletics Women's 4 x 400 metres Relay	Bronz
23849	135563	Olesya Nikolayevna Zykina	F	23	171	64.0	Russia	RUS	2004 Summer	2004	Summer	Athina	Athletics	Athletics Women's 4 x 400 metres Relay	Silv

1239 rows × 15 columns

In [321…

```
fdata['event'].unique()
```

Out[321…

```
array(["Athletics Women's Javelin Throw",
       "Athletics Women's 5,000 metres", "Athletics Women's Shot Put",
       "Athletics Women's 4 x 400 metres Relay",
       "Athletics Women's Long Jump",
       "Athletics Women's 4 x 100 metres Relay",
       "Athletics Women's 100 metres Hurdles",
       "Athletics Women's 400 metres", "Athletics Women's Marathon",
       "Athletics Women's Discus Throw",
       "Athletics Women's 400 metres Hurdles",
       "Athletics Women's 100 metres",
       "Athletics Women's 3,000 metres Steeplechase",
       "Athletics Women's High Jump", "Athletics Women's 10,000 metres",
       "Athletics Women's 1,500 metres", "Athletics Women's Heptathlon",
       "Athletics Women's 800 metres", "Athletics Women's Pentathlon",
       "Athletics Women's 200 metres",
       "Athletics Women's 80 metres Hurdles",
       "Athletics Women's 3,000 metres",
       "Athletics Women's 10 kilometres Walk",
       "Athletics Women's Hammer Throw", "Athletics Women's Triple Jump",
       "Athletics Women's Pole Vault",
       "Athletics Women's 20 kilometres Walk"], dtype=object)
```

In [327…

```python
fdata100 = fdata[fdata['event'].str.contains('100 metres Hurdles')]
fdata100
```

Out[327…

	id	name	sex	age	height	weight	team	noc	games	year	season	city	sport	event
329	2695	Nia Sifaatihii Ali	F	27	170	65.0	United States	USA	2016 Summer	2016	Summer	Rio de Janeiro	Athletics	Athletics Women's 100 metres Hurdles
366	3055	Glory Alozie Oluchi	F	22	155	51.0	Nigeria	NGR	2000 Summer	2000	Summer	Sydney	Athletics	Athletics Women's 100 metres Hurdles
581	4390	Tatyana Mikhaylovna Anisimova (Poluboyarova-)	F	26	172	65.0	Soviet Union	URS	1976 Summer	1976	Summer	Montreal	Athletics	Athletics Women's 100 metres Hurdles
2588	16188	Valeria Bufanu-tefnescu	F	25	170	60.0	Romania	ROU	1972 Summer	1972	Summer	Munich	Athletics	Athletics Women's 100 metres Hurdles
2610	16305	Brigita Bukovec	F	26	168	55.0	Slovenia	SLO	1996 Summer	1996	Summer	Atlanta	Athletics	Athletics Women's 100 metres Hurdles
3019	18976	Kristi Castlin	F	28	170	60.0	United States	USA	2016 Summer	2016	Summer	Rio de Janeiro	Athletics	Athletics Women's 100 metres Hurdles
3127	19878	Michle Marie George Chardonnet (-Piasenta)	F	27	170	60.0	France	FRA	1984 Summer	1984	Summer	Los Angeles	Athletics	Athletics Women's 100 metres Hurdles
4688	29394	Yordanka Lyubchova Donkova	F	26	175	67.0	Bulgaria	BUL	1988 Summer	1988	Summer	Seoul	Athletics	Athletics Women's 100 metres Hurdles
4689	29394	Yordanka Lyubchova Donkova	F	30	175	67.0	Bulgaria	BUL	1992 Summer	1992	Summer	Barcelona	Athletics	Athletics Women's 100 metres Hurdles
5113	31652	Anneliese Ehrhardt (Jahns-)	F	22	166	58.0	East Germany	GDR	1972 Summer	1972	Summer	Munich	Athletics	Athletics Women's 100 metres Hurdles
5769	35644	Benita P. Fitzgerald-Brown (-Mosley)	F	23	178	65.0	United States	USA	1984 Summer	1984	Summer	Los Angeles	Athletics	Athletics Women's 100 metres Hurdles

6625	40533	Patricia Girard-Lno	F	28	164	51.0	France	FRA	1996 Summer	1996	Summer	Atlanta	Athletics	Athletics Women's 100 metres Hurdles
7598	46113	Dawn Harper (-Everhart, -Nelson)	F	24	168	61.0	United States	USA	2008 Summer	2008	Summer	Beijing	Athletics	Athletics Women's 100 metres Hurdles
7599	46113	Dawn Harper (-Everhart, -Nelson)	F	28	168	61.0	United States	USA	2012 Summer	2012	Summer	London	Athletics	Athletics Women's 100 metres Hurdles
7702	46845	Joanna Dove Hayes	F	27	167	58.0	United States	USA	2004 Summer	2004	Summer	Athina	Athletics	Athletics Women's 100 metres Hurdles
10540	62331	Vera Yakovlevna Komisova (Nikitina-)	F	27	169	60.0	Soviet Union	URS	1980 Summer	1980	Summer	Moskva	Athletics	Athletics Women's 100 metres Hurdles
11389	66456	Lucyna Langer (-Kaek)	F	24	167	55.0	Poland	POL	1980 Summer	1980	Summer	Moskva	Athletics	Athletics Women's 100 metres Hurdles
11625	67624	Nataliya Vasilyevna Lebedeva (-Suprun, -Monast...	F	26	171	62.0	Soviet Union	URS	1976 Summer	1976	Summer	Montreal	Athletics	Athletics Women's 100 metres Hurdles
12277	71312	Priscilla Lopes-Schliep	F	25	163	67.0	Canada	CAN	2008 Summer	2008	Summer	Beijing	Athletics	Athletics Women's 100 metres Hurdles
12998	75590	LaVonna Ann Martin (-Floral)	F	25	170	66.0	United States	USA	1992 Summer	1992	Summer	Barcelona	Athletics	Athletics Women's 100 metres Hurdles
13411	77809	Sally McLellan-Pearson	F	21	167	60.0	Australia	AUS	2008 Summer	2008	Summer	Beijing	Athletics	Athletics Women's 100 metres Hurdles
13412	77809	Sally McLellan-Pearson	F	25	167	60.0	Australia	AUS	2012 Summer	2012	Summer	London	Athletics	Athletics Women's 100 metres Hurdles
14141	82292	Melissa Renee Morrison (-Howard)	F	29	164	56.0	United States	USA	2000 Summer	2000	Summer	Sydney	Athletics	Athletics Women's 100 metres Hurdles
14142	82292	Melissa Renee Morrison (-Howard)	F	33	164	56.0	United States	USA	2004 Summer	2004	Summer	Athina	Athletics	Athletics Women's 100 metres Hurdles
14528	84563	Ludmila Viktorovna Narozhilenko-Engquist (Leon...	F	32	174	66.0	Sweden	SWE	1996 Summer	1996	Summer	Atlanta	Athletics	Athletics Women's 100 metres Hurdles
15482	90156	Olena Anatolivna Ovcharova-Krasovska	F	27	176	62.0	Ukraine	UKR	2004 Summer	2004	Summer	Athina	Athletics	Athletics Women's 100 metres Hurdles
15720	92060	Paraskevi "Voula" Patoulidou (-Zarzavatsidou)	F	27	167	60.0	Greece	GRE	1992 Summer	1992	Summer	Barcelona	Athletics	Athletics Women's 100 metres Hurdles
														Athletics

17274	100577	Karin Richert-Balzer	F	34	171	64.0	East Germany	GDR	1972 Summer	1972	Summer	Munich	Athletics	Women's 100 metres Hurdles
17561	102175	Brianna Rollins	F	24	165	59.0	United States	USA	2016 Summer	2016	Summer	Rio de Janeiro	Athletics	Athletics Women's 100 metres Hurdles
18321	106582	Johanna Schaller-Klier	F	23	176	70.0	East Germany	GDR	1976 Summer	1976	Summer	Montreal	Athletics	Athletics Women's 100 metres Hurdles
18322	106582	Johanna Schaller-Klier	F	27	176	70.0	East Germany	GDR	1980 Summer	1980	Summer	Moskva	Athletics	Athletics Women's 100 metres Hurdles
19004	109895	Olga Vasilyevna Shishigina (-Kutsakova)	F	31	165	57.0	Kazakhstan	KAZ	2000 Summer	2000	Summer	Sydney	Athletics	Athletics Women's 100 metres Hurdles
19078	110247	Gloria Siebert (Kovarik-, Uibel-)	F	24	170	54.0	East Germany	GDR	1988 Summer	1988	Summer	Seoul	Athletics	Athletics Women's 100 metres Hurdles
20163	115848	Shirley Elaine Strong	F	25	173	63.0	Great Britain	GBR	1984 Summer	1984	Summer	Los Angeles	Athletics	Athletics Women's 100 metres Hurdles
21448	122788	Kimberly "Kim" Turner (-McKenzie)	F	23	165	57.0	United States	USA	1984 Summer	1984	Summer	Los Angeles	Athletics	Athletics Women's 100 metres Hurdles
22658	129536	Kellie Wells	F	30	163	57.0	United States	USA	2012 Summer	2012	Summer	London	Athletics	Athletics Women's 100 metres Hurdles
23469	133635	Claudia Hildegard Zaczkiewicz (Reidick-, -Reid...	F	26	170	57.0	West Germany	FRG	1988 Summer	1988	Summer	Seoul	Athletics	Athletics Women's 100 metres Hurdles

There is no event that is '100 metres sprint' so in that case I am using '100 metres hurdles'

In [330…

```
_95th_percentile = fdata100['height'].quantile(0.95)
print('The 95th percentile for the height column is',_95th_percentile)
```

```
The 95th percentile for the height column is 176.0
```

In [ ]:

QUESTION: Find the single woman’s event that depicts the highest variability in the height of the competitor across the entire history of Olympics.

In [333…

```
### Generating a new set of data
fdata2 = df3[df3['sex'] == 'F']
fdata2
```

Out[333…

	id	name	sex	age	height	weight	team	noc	games	year	season	city	sport	event
7	37	Ann Kristin Aarnes	F	23	182	64.0	Norway	NOR	1996 Summer	1996	Summer	Atlanta	Football	Football Women's Football
10	67	Mariya Vasilyevna Abakumova (-Tarabina)	F	22	179	80.0	Russia	RUS	2008 Summer	2008	Summer	Beijing	Athletics	Athletics Women's Javelin Throw
20	90	Tamila Rashidovna Abasova	F	21	163	60.0	Russia	RUS	2004 Summer	2004	Summer	Athina	Cycling	Cycling Women's Sprint
34	259	Reema Abdo	F	21	173	59.0	Canada	CAN	1984 Summer	1984	Summer	Los Angeles	Swimming	Swimming Women's 4 x 100 metres Medley Relay
46	394	Irene Abel	F	19	160	48.0	East Germany	GDR	1972 Summer	1972	Summer	Munich	Gymnastics	Gymnastics Women's Team All-Around
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
23844	135553	Galina Ivanovna Zybina (-Fyodorova)	F	21	168	80.0	Soviet Union	URS	1952 Summer	1952	Summer	Helsinki	Athletics	Athletics Women's Shot Put
23845	135553	Galina Ivanovna Zybina (-Fyodorova)	F	25	168	80.0	Soviet Union	URS	1956 Summer	1956	Summer	Melbourne	Athletics	Athletics Women's Shot Put
23846	135553	Galina Ivanovna Zybina (-Fyodorova)	F	33	168	80.0	Soviet Union	URS	1964 Summer	1964	Summer	Tokyo	Athletics	Athletics Women's Shot Put
23848	135563	Olesya Nikolayevna Zykina	F	19	171	64.0	Russia	RUS	2000 Summer	2000	Summer	Sydney	Athletics	Athletics Women's 4 x 400 metres Relay
23849	135563	Olesya Nikolayevna Zykina	F	23	171	64.0	Russia	RUS	2004 Summer	2004	Summer	Athina	Athletics	Athletics Women's 4 x 400 metres Relay

7790 rows × 15 columns

In [336…

```
sd_fdata2  = fdata2.groupby('event')['height'].std().reset_index()
sd_fdata2
```

Out[336…

	event	height
0	Athletics Women's 1,500 metres	5.032908
1	Athletics Women's 10 kilometres Walk	4.308906
2	Athletics Women's 10,000 metres	5.409697
3	Athletics Women's 100 metres	6.289850
4	Athletics Women's 100 metres Hurdles	4.677272
...	...	...
127	Wrestling Women's Flyweight, Freestyle	2.924445
128	Wrestling Women's Heavyweight, Freestyle	4.250490
129	Wrestling Women's Light-Heavyweight, Freestyle	6.658328
130	Wrestling Women's Lightweight, Freestyle	3.976119
131	Wrestling Women's Middleweight, Freestyle	4.043808

132 rows × 2 columns

In [340…

```
sd_fdata2.sort_values('height',ascending=False)
```

Out[340…

	event	height
84	Rowing Women's Coxed Fours	10.865490
27	Basketball Women's Basketball	9.700255
85	Rowing Women's Coxed Quadruple Sculls	9.246396
83	Rowing Women's Coxed Eights	8.741931
110	Swimming Women's 100 metres Butterfly	8.134399
...	...	...
127	Wrestling Women's Flyweight, Freestyle	2.924445
30	Boxing Women's Middleweight	2.911390
92	Sailing Mixed Multihull	2.081666
98	Shooting Mixed Skeet	NaN
99	Shooting Mixed Small-Bore Rifle, Three Positio...	NaN

132 rows × 2 columns

The answer to the question is Rowing Women's Coxed Fours

In [ ]:

QUESTION: How the average age of swimmers in Olympic has evolved with time. How has this changed over time?

In [341…

```
df3.columns
```

Out[341…

```
Index(['id', 'name', 'sex', 'age', 'height', 'weight', 'team', 'noc', 'games',
       'year', 'season', 'city', 'sport', 'event', 'medal'],
      dtype='object')
```

In [ ]:

In [357…

```
mdata2 = df3[df3['sex'] == 'M']
mdata = mdata2.groupby('year')['age'].mean().reset_index()
mdata
```

Out[357…

	year	age
0	1896	23.277778
1	1900	23.542857
2	1904	23.481481
3	1906	26.561404
4	1908	24.280303
5	1912	23.937500
6	1920	25.572289
7	1924	25.285714
8	1928	24.788732
9	1932	24.117241
10	1936	24.789157
11	1948	26.751553
12	1952	26.021097
13	1956	25.587838
14	1960	25.516779
15	1964	25.904306
16	1968	25.605956
17	1972	25.439126
18	1976	25.211329
19	1980	25.063694
20	1984	24.881321
21	1988	25.264000
22	1992	25.627535
23	1994	25.764706
24	1996	26.065668
25	1998	27.564706
26	2000	26.639471
27	2002	29.430108
28	2004	26.755605
29	2006	29.326733
30	2008	26.613408
31	2010	28.052083
32	2012	26.687861
33	2014	28.564356
34	2016	27.061700

In [358…

```
fdata = fdata2.groupby('year')['age'].mean().reset_index()
fdata
```

Out[358…

	year	age
0	1924	18.000000
1	1928	21.526316
2	1932	20.416667
3	1936	22.409091
4	1948	25.380952
5	1952	24.066667
6	1956	22.436170
7	1960	22.357143
8	1964	22.901961
9	1968	22.261261
10	1972	22.431373
11	1976	22.446970
12	1980	22.731207
13	1984	23.144309
14	1988	23.264368
15	1992	24.303493
16	1994	28.916667
17	1996	25.194707
18	1998	25.797619
19	2000	26.150388
20	2002	25.367816
21	2004	25.959276
22	2006	25.933333
23	2008	25.919473
24	2010	25.755556
25	2012	26.000000
26	2014	25.373494
27	2016	26.103683

In [359…

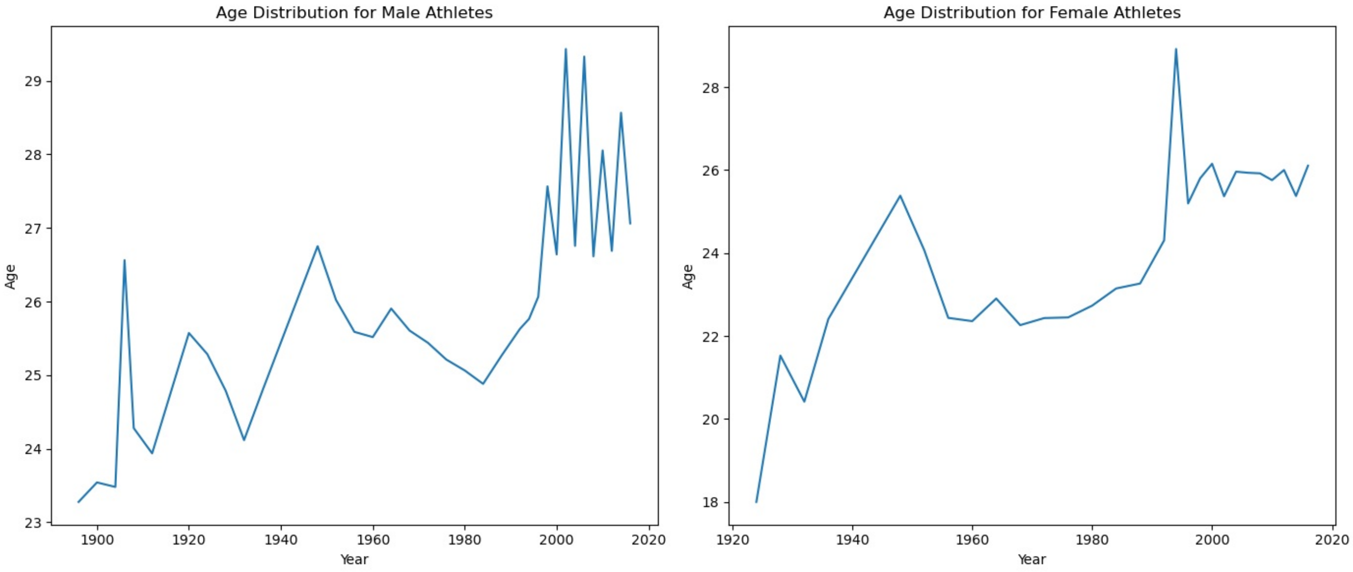
```python
fig, axes = plt.subplots(1, 2, figsize=(14, 6))

# First subplot for male athletes
sns.lineplot(x='year', y='age', data=mdata, ax=axes[0])
axes[0].set_title('Age Distribution for Male Athletes')
axes[0].set_xlabel('Year')
axes[0].set_ylabel('Age')

# Second subplot for female athletes
sns.lineplot(x='year', y='age', data=fdata, ax=axes[1])
axes[1].set_title('Age Distribution for Female Athletes')
axes[1].set_xlabel('Year')
axes[1].set_ylabel('Age')


plt.tight_layout()
plt.show()
```

```
C:\Users\tegbe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is depr
ecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\tegbe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is depr
ecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\tegbe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is depr
ecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\tegbe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is depr
ecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



- From the Line plots above, we can see that average age over the years was in an upward trend for both Genders
- However among the Male athletes, from 1960 to 1980 the avereage age decreased meaning younger male athletes competed. This is not the case among Female athletes during the period.
- The female average age dropped in 1997-8

In [ ]: