
1. Write a program to initialize a dictionary (of your choice) and create a pandas DataFrame from it. Print the dataframe?

```
In [1]: #Code here
import pandas as pd

data = {
    'Name': ['Alice', 'Bob', 'Charlie'],
    'Age': [25, 30, 35],
    'City': ['New York', 'London', 'Paris']
}
df_dict = pd.DataFrame(data)
print(df_dict)
```

	Name	Age	City
0	Alice	25	New York
1	Bob	30	London
2	Charlie	35	Paris

2. Write a program to read the CSV file "summer.csv" into a pandas DataFrame "df", display the first 5 rows, and print the DataFrame's shape

```
In [2]: #Code here
df = pd.read_csv('summer.csv')
print(df.head())
print("Shape of DataFrame:", df.shape)
```

	Year	City	Sport	Discipline	Athlete	Country	Gender	\
0	1896	Athens	Aquatics	Swimming	HAJOS, Alfred	HUN	Men	
1	1896	Athens	Aquatics	Swimming	HERSCHMANN, Otto	AUT	Men	
2	1896	Athens	Aquatics	Swimming	DRIVAS, Dimitrios	GRE	Men	
3	1896	Athens	Aquatics	Swimming	MALOKINIS, Ioannis	GRE	Men	
4	1896	Athens	Aquatics	Swimming	CHASAPIS, Spiridon	GRE	Men	

		Event	Medal
0		100M Freestyle	Gold
1		100M Freestyle	Silver
2	100M Freestyle	For Sailors	Bronze
3	100M Freestyle	For Sailors	Gold
4	100M Freestyle	For Sailors	Silver

Shape of DataFrame: (31165, 9)

3. Write a program to display the last 10 rows of the DataFrame?

```
In [3]: #Code here
print(df.tail(10))
```

	Year	City	Sport	Discipline \
31155	2012	London	Wrestling	Wrestling Freestyle
31156	2012	London	Wrestling	Wrestling Freestyle
31157	2012	London	Wrestling	Wrestling Freestyle
31158	2012	London	Wrestling	Wrestling Freestyle
31159	2012	London	Wrestling	Wrestling Freestyle
31160	2012	London	Wrestling	Wrestling Freestyle
31161	2012	London	Wrestling	Wrestling Freestyle
31162	2012	London	Wrestling	Wrestling Freestyle
31163	2012	London	Wrestling	Wrestling Freestyle
31164	2012	London	Wrestling	Wrestling Freestyle

	Athlete	Country	Gender	Event	Medal
31155	AHMADOV, Emin	AZE	Men	Wg 74 KG	Bronze
31156	KAZAKEVIC, Aleksandr	LTU	Men	Wg 74 KG	Bronze
31157	KHUGAEV, Alan	RUS	Men	Wg 84 KG	Gold
31158	EBRAHIM, Karam Mohamed Gaber	EGY	Men	Wg 84 KG	Silver
31159	GAJIYEV, Danyal	KAZ	Men	Wg 84 KG	Bronze
31160	JANIKOWSKI, Damian	POL	Men	Wg 84 KG	Bronze
31161	REZAEI, Ghasem Gholamreza	IRI	Men	Wg 96 KG	Gold
31162	TOTROV, Rustam	RUS	Men	Wg 96 KG	Silver
31163	ALEKSANYAN, Artur	ARM	Men	Wg 96 KG	Bronze
31164	LIDBERG, Jimmy	SWE	Men	Wg 96 KG	Bronze

4. Write a program to get a summary of the DataFrame (such as the mean, count, etc.) and print the data types of all columns?

```
In [4]: #Code here
print(df.describe(include='all'))
print("Data Types:\n", df.dtypes)
```

	Year	City	Sport	Discipline	Athlete	Country \
count	31165.000000	31165	31165	31165	31165	31161
unique	NaN	22	43	67	22762	147
top	NaN	London	Aquatics	Athletics	PHELPS, Michael	USA
freq	NaN	3567	4170	3638	22	4585
mean	1970.482785	NaN	NaN	NaN	NaN	NaN
std	33.158416	NaN	NaN	NaN	NaN	NaN
min	1896.000000	NaN	NaN	NaN	NaN	NaN
25%	1948.000000	NaN	NaN	NaN	NaN	NaN
50%	1980.000000	NaN	NaN	NaN	NaN	NaN
75%	2000.000000	NaN	NaN	NaN	NaN	NaN
max	2012.000000	NaN	NaN	NaN	NaN	NaN

	Gender	Event	Medal
count	31165	31165	31165
unique	2	666	3
top	Men	Football	Gold
freq	22746	1497	10486
mean	NaN	NaN	NaN
std	NaN	NaN	NaN
min	NaN	NaN	NaN
25%	NaN	NaN	NaN
50%	NaN	NaN	NaN
75%	NaN	NaN	NaN
max	NaN	NaN	NaN

Data Types:

Year	int64
City	object
Sport	object
Discipline	object
Athlete	object
Country	object
Gender	object
Event	object
Medal	object

dtype: object

5. Write a program to select a single column "Gender" from the DataFrame?

```
In [5]: #Code here
gender_column = df['Gender']
print(gender_column)
```

```
0      Men
1      Men
2      Men
3      Men
4      Men
...
31160   Men
31161   Men
31162   Men
31163   Men
31164   Men
Name: Gender, Length: 31165, dtype: object
```

6. Write a program to select multiple columns "Year" and "City" from the DataFrame?

```
In [6]: #Code here
year_city = df[['Year', 'City']]
print(year_city)
```

```
      Year  City
0      1896 Athens
1      1896 Athens
2      1896 Athens
3      1896 Athens
4      1896 Athens
...      ...   ...
31160  2012 London
31161  2012 London
31162  2012 London
31163  2012 London
31164  2012 London
```

```
[31165 rows x 2 columns]
```

7. Write a program to filter rows in a DataFrame based on the condition that Gender = Men ?

```
In [7]: #Code here
men_df = df[df['Gender'] == 'Men']
print(men_df)
```

	Year	City	Sport	Discipline \
0	1896	Athens	Aquatics	Swimming
1	1896	Athens	Aquatics	Swimming
2	1896	Athens	Aquatics	Swimming
3	1896	Athens	Aquatics	Swimming
4	1896	Athens	Aquatics	Swimming
...
31160	2012	London	Wrestling	Wrestling Freestyle
31161	2012	London	Wrestling	Wrestling Freestyle
31162	2012	London	Wrestling	Wrestling Freestyle
31163	2012	London	Wrestling	Wrestling Freestyle
31164	2012	London	Wrestling	Wrestling Freestyle

	Athlete	Country	Gender	Event \
0	HAJOS, Alfred	HUN	Men	100M Freestyle
1	HERSCHMANN, Otto	AUT	Men	100M Freestyle
2	DRIVAS, Dimitrios	GRE	Men	100M Freestyle For Sailors
3	MALOKINIS, Ioannis	GRE	Men	100M Freestyle For Sailors
4	CHASAPIS, Spiridon	GRE	Men	100M Freestyle For Sailors
...
31160	JANIKOWSKI, Damian	POL	Men	Wg 84 KG
31161	REZAEI, Ghasem Gholamreza	IRI	Men	Wg 96 KG
31162	TOTROV, Rustam	RUS	Men	Wg 96 KG
31163	ALEKSANYAN, Artur	ARM	Men	Wg 96 KG
31164	LIDBERG, Jimmy	SWE	Men	Wg 96 KG

	Medal
0	Gold
1	Silver
2	Bronze
3	Gold
4	Silver
...	...
31160	Bronze
31161	Gold
31162	Silver
31163	Bronze
31164	Bronze

[22746 rows x 9 columns]

8. Write a program to select and print the first 5 rows and columns 2 to 6 (inclusive) of the DataFrame. Also, print the type of the resulting object?

```
In [8]: #Code here
subset = df.iloc[:5, 2:7]
print(subset)
print("Type:", type(subset))
```

	Sport	Discipline	Athlete	Country	Gender
0	Aquatics	Swimming	HAJOS, Alfred	HUN	Men
1	Aquatics	Swimming	HERSCHMANN, Otto	AUT	Men
2	Aquatics	Swimming	DRIVAS, Dimitrios	GRE	Men
3	Aquatics	Swimming	MALOKINIS, Ioannis	GRE	Men
4	Aquatics	Swimming	CHASAPIS, Spiridon	GRE	Men

Type: <class 'pandas.core.frame.DataFrame'>

9. Write a program to delete a column "City" from the DataFrame and print the new dataframe??

```
In [9]: #Code here
df_dropped = df.drop(columns=['City'])
print(df_dropped.head())
```

	Year	Sport	Discipline	Athlete	Country	Gender	\
0	1896	Aquatics	Swimming	HAJOS, Alfred	HUN	Men	
1	1896	Aquatics	Swimming	HERSCHMANN, Otto	AUT	Men	
2	1896	Aquatics	Swimming	DRIVAS, Dimitrios	GRE	Men	
3	1896	Aquatics	Swimming	MALOKINIS, Ioannis	GRE	Men	
4	1896	Aquatics	Swimming	CHASAPIS, Spiridon	GRE	Men	

	Event	Medal
0	100M Freestyle	Gold
1	100M Freestyle	Silver
2	100M Freestyle For Sailors	Bronze
3	100M Freestyle For Sailors	Gold
4	100M Freestyle For Sailors	Silver

10. Write a program to rename a column "Athlete" to "Participants" in the DataFrame?

```
In [10]: #Code here
df_renamed = df.rename(columns={'Athlete': 'Participants'})
print(df_renamed.head())
```

	Year	City	Sport	Discipline	Participants	Country	Gender	\
0	1896	Athens	Aquatics	Swimming	HAJOS, Alfred	HUN	Men	
1	1896	Athens	Aquatics	Swimming	HERSCHMANN, Otto	AUT	Men	
2	1896	Athens	Aquatics	Swimming	DRIVAS, Dimitrios	GRE	Men	
3	1896	Athens	Aquatics	Swimming	MALOKINIS, Ioannis	GRE	Men	
4	1896	Athens	Aquatics	Swimming	CHASAPIS, Spiridon	GRE	Men	

	Event	Medal
0	100M Freestyle	Gold
1	100M Freestyle	Silver
2	100M Freestyle For Sailors	Bronze
3	100M Freestyle For Sailors	Gold
4	100M Freestyle For Sailors	Silver

11. Write a program to select and print the "Participants", "Medal" and "Event" columns for index 10 and 201 from the DataFrame.?

```
In [11]: #Code here
subset = df.loc[[10, 201], ['Athlete', 'Medal', 'Event']]
print(subset)
```

	Athlete	Medal	Event
10	PEPANOS, Antonios	Silver	400M Freestyle
201	LISTER, William	Gold	Water Polo

12. Write a program to display the first 10 rows of the df DataFrame and check for any missing values in each column.?

```
In [12]: #Code here
print(df.head(10))
print("Missing Values:\n", df.isnull().sum())
```

	Year	City	Sport	Discipline	Athlete	Country	Gender	\
0	1896	Athens	Aquatics	Swimming	HAJOS, Alfred	HUN	Men	
1	1896	Athens	Aquatics	Swimming	HERSCHMANN, Otto	AUT	Men	
2	1896	Athens	Aquatics	Swimming	DRIVAS, Dimitrios	GRE	Men	
3	1896	Athens	Aquatics	Swimming	MALOKINIS, Ioannis	GRE	Men	
4	1896	Athens	Aquatics	Swimming	CHASAPIS, Spiridon	GRE	Men	
5	1896	Athens	Aquatics	Swimming	CHOROPHAS, Efstathios	GRE	Men	
6	1896	Athens	Aquatics	Swimming	HAJOS, Alfred	HUN	Men	
7	1896	Athens	Aquatics	Swimming	ANDREOU, Joannis	GRE	Men	
8	1896	Athens	Aquatics	Swimming	CHOROPHAS, Efstathios	GRE	Men	
9	1896	Athens	Aquatics	Swimming	NEUMANN, Paul	AUT	Men	

	Event	Medal
0	100M Freestyle	Gold
1	100M Freestyle	Silver
2	100M Freestyle For Sailors	Bronze
3	100M Freestyle For Sailors	Gold
4	100M Freestyle For Sailors	Silver
5	1200M Freestyle	Bronze
6	1200M Freestyle	Gold
7	1200M Freestyle	Silver
8	400M Freestyle	Bronze
9	400M Freestyle	Gold

Missing Values:

Year	0
City	0
Sport	0
Discipline	0
Athlete	0
Country	4
Gender	0
Event	0
Medal	0

dtype: int64

13. Write a program to select and print the "Country" and "Medal" columns from the df DataFrame, and then print the number of unique values in each of these columns.?

```
In [13]: #Code here
country_medal = df[['Country', 'Medal']]
print(country_medal.head())
print("Unique Countries:", df['Country'].nunique())
print("Unique Medals:", df['Medal'].nunique())
```

```
Country  Medal
0      HUN   Gold
1      AUT  Silver
2      GRE  Bronze
3      GRE   Gold
4      GRE  Silver
Unique Countries: 147
Unique Medals: 3
```

14. Write a program to group the df DataFrame by the "Country" column and calculate the total number of medals for each country. Print the result.?

```
In [14]: #Code here
medal_counts = df.groupby('Country')['Medal'].count()
print(medal_counts)
```

```
Country
AFG      2
AHO      1
ALG     15
ANZ     29
ARG    259
...
VIE      2
YUG    435
ZAM      2
ZIM     23
ZZX     48
Name: Medal, Length: 147, dtype: int64
```

15. Write a program to sort the df DataFrame by the "Year" column in descending order and print the first 5 rows of the sorted DataFrame.?

```
In [15]: #Code here
df_sorted = df.sort_values(by='Year', ascending=False)
print(df_sorted.head())
```


	Year	City	Sport	Discipline	Athlete \
31164	2012	London	Wrestling	Wrestling Freestyle	LIDBERG, Jimmy
29885	2012	London	Boxing	Boxing	NEVIN, John Joe
29857	2012	London	Basketball	Basketball	BISHOP, Abby
29858	2012	London	Basketball	Basketball	CAMBAGE, Elizabeth
29859	2012	London	Basketball	Basketball	HARROWER, Kristi

	Country	Gender	Event	Medal
31164	SWE	Men	Wg 96 KG	Bronze
29885	IRL	Men	56KG	Silver
29857	AUS	Women	Basketball	Bronze
29858	AUS	Women	Basketball	Bronze
29859	AUS	Women	Basketball	Bronze

16. Write a program to add a new column "Century" to the df DataFrame. Classify each event as occurring in the "20th Century" if the "Year" is less than 2000, or the "21st Century" if the "Year" is 2000 or later. Print the first 5 rows to verify the change.

```
In [16]: #Code here
df['Century'] = df['Year'].apply(lambda x: '20th Century' if x < 2000 else '21st Century')
print(df[['Year', 'Century']].head())
```

	Year	Century
0	1896	20th Century
1	1896	20th Century
2	1896	20th Century
3	1896	20th Century
4	1896	20th Century

17. Write a program to filter and print rows from the df DataFrame where the "Country" is "USA" and the "Medal" is "Gold". Also, count the number of rows that meet this condition.

```
In [17]: #Code here
usa_gold = df[(df['Country'] == 'USA') & (df['Medal'] == 'Gold')]
print(usa_gold)
print("Number of Gold Medals by USA:", len(usa_gold))
```

	Year	City	Sport	Discipline \
13	1896	Athens	Athletics	Athletics
15	1896	Athens	Athletics	Athletics
21	1896	Athens	Athletics	Athletics
27	1896	Athens	Athletics	Athletics
29	1896	Athens	Athletics	Athletics
...
30955	2012	London	Tennis	Tennis
30970	2012	London	Volleyball	Beach Volleyball
30971	2012	London	Volleyball	Beach Volleyball
31125	2012	London	Wrestling	Wrestling Freestyle
31133	2012	London	Wrestling	Wrestling Freestyle

	Athlete	Country	Gender	Event	Medal \
13	BURKE, Thomas	USA	Men	100M	Gold
15	CURTIS, Thomas	USA	Men	110M Hurdles	Gold
21	BURKE, Thomas	USA	Men	400M	Gold
27	GARRETT, Robert	USA	Men	Discus Throw	Gold
29	CLARK, Ellery	USA	Men	High Jump	Gold
...
30955	WILLIAMS, Serena	USA	Women	Singles	Gold
30970	MAY, Misty	USA	Women	Beach Volleyball	Gold
30971	WALSH JENNINGS, Kerri	USA	Women	Beach Volleyball	Gold
31125	BURROUGHS, Jordan Ernest	USA	Men	Wf 74 KG	Gold
31133	VARNER, Jacob Stephen	USA	Men	Wf 96 KG	Gold

	Century
13	20th Century
15	20th Century
21	20th Century
27	20th Century
29	20th Century
...	...
30955	21st Century
30970	21st Century
30971	21st Century
31125	21st Century
31133	21st Century

[2235 rows x 10 columns]

Number of Gold Medals by USA: 2235

18. Write a program to create a new column "Medal Points" in the df DataFrame. Assign 3 points for "Gold", 2 points for "Silver", and 1 point for "Bronze". Print the first 5 rows to verify the changes.

```
In [18]: #Code here
medal_points_map = {'Gold': 3, 'Silver': 2, 'Bronze': 1}
df['Medal Points'] = df['Medal'].map(medal_points_map)
print(df[['Athlete', 'Medal', 'Medal Points']].head())
```

	Athlete	Medal	Medal Points
0	HAJOS, Alfred	Gold	3
1	HERSCHMANN, Otto	Silver	2
2	DRIVAS, Dimitrios	Bronze	1
3	MALOKINIS, Ioannis	Gold	3
4	CHASAPIS, Spiridon	Silver	2

19. Write a program to filter the df DataFrame to include only rows where the "Country" is "USA". Then, group the filtered DataFrame by "Year" and calculate the total number of medals won by the USA each year. Print the results.

```
In [19]: #Code here
usa_df = df[df['Country'] == 'USA']
medals_per_year = usa_df.groupby('Year')['Medal'].count()
print(medals_per_year)
```

```
Year
1896      20
1900      55
1904     394
1908      63
1912     101
1920     193
1924     198
1928      84
1932     181
1936      92
1948     148
1952     130
1956     118
1960     112
1964     150
1968     149
1972     155
1976     155
1984     333
1988     193
1992     224
1996     260
2000     248
2004     264
2008     315
2012     250
Name: Medal, dtype: int64
```

20. Write a program to identify the athlete who has won the most medals. Display the athlete's name and the total number of medals they have won.

```
In [20]: #Code here
top_athlete = df['Athlete'].value_counts().idxmax()
```

```
top_count = df['Athlete'].value_counts().max()
print(f"Athlete with most medals: {top_athlete} ({top_count} medals)")
```

Athlete with most medals: PHELPS, Michael (22 medals)

21. Write a program to group the df DataFrame by the "Sport" column and calculate the number of each type of medal (Gold, Silver, Bronze) for each sport. Print the results.

```
In [21]: #Code here
sport_medals = df.groupby(['Sport', 'Medal']).size().unstack(fill_value=0)
print(sport_medals)
```

Medal Sport	Bronze	Gold	Silver
Aquatics	1365	1421	1384
Archery	90	126	113
Athletics	1199	1215	1224
Badminton	52	46	46
Baseball	112	111	112
Basketball	334	342	336
Basque Pelota	0	2	2
Boxing	416	239	239
Canoe	27	27	27
Canoe / Kayak	334	334	334
Cricket	0	12	12
Croquet	2	4	2
Cycling	365	371	369
Equestrian	306	320	313
Fencing	519	552	542
Football	509	503	485
Golf	4	13	13
Gymnastics	731	820	756
Handball	323	321	329
Hockey	459	486	477
Ice Hockey	8	8	11
Jeu de paume	1	1	1
Judo	246	123	122
Lacrosse	0	28	31
Modern Pentathlon	60	60	60
Polo	17	22	27
Rackets	4	3	3
Roque	1	1	1
Rowing	888	890	889
Rugby	23	78	91
Sailing	325	412	372
Shooting	380	384	386
Skating	9	9	9
Softball	60	60	60
Table Tennis	52	46	46
Taekwondo	48	32	32
Tennis	114	92	90
Triathlon	8	8	8
Tug of War	22	36	36
Volleyball	331	332	331
Water Motorsports	0	5	0
Weightlifting	197	200	196
Wrestling	428	391	393

Good Luck!