EDS ACTIVITY:

NAME: SIDDHESH ABASAHEB MANE
ROLL NO.: CS1-06
PRN: 202401040187
DATASET: FIFA DATASET

Importing dataset:

```
import pandas as pd
import numpy as np

df = pd.read_csv('FIFA23_official_data.csv')
```

1. Average overall rating of all players

```
average_overall = df['Overall'].mean()
print("Average Overall Rating:", average_overall)
```

```
Average Overall Rating: 63.36959229898075
```

2. Player with the highest potential

```
highest_potential_player = df.loc[df['Potential'].idxmax()]
print(highest_potential_player[['Name', 'Potential']])
```

```
Name         K. Mbappé
Potential           95
Name: 75, dtype: object
```

3. Median age of all players

```
median_age = df['Age'].median()
print("Median Age:", median_age)
```

```
Median Age: 22.0
```

4. Club with the maximum number of players

```
top_club = df['Club'].value_counts().idxmax()
print("Club with Maximum Players:", top_club)
```

```
Club with Maximum Players: Manchester United
```

5. Top 5 players with the highest wages

```
top_5_wages = df[['Name', 'Wage']].sort_values(by='Wage', ascending=False).head(5)
print(top_5_wages)
```

```
              Name Wage
1677      L. Reis  €9K
5129     K. Bryan  €9K
5021   H. Darling  €9K
3127  R. Petretta  €9K
1216    F. Farías  €9K
```

6. Total market value of all players

```
def value_to_float(val):
    val = val.replace('€','')
    if 'M' in val:
        return float(val.replace('M','')) * 1e6
```

```
    elif 'K' in val:
        return float(val.replace('K','')) * 1e3
    else:
        return float(val)

df['Value_num'] = df['Value'].apply(value_to_float)
total_market_value = df['Value_num'].sum()
print("Total Market Value: €", total_market_value)
```

```
Total Market Value: € 48384663000.0
```

7. Youngest player and his nationality

```
youngest_player = df.loc[df['Age'].idxmin()]
print(youngest_player[['Name', 'Nationality', 'Age']])
```

```
Name           22 D. Oncescu
Nationality          Romania
Age                       15
Name: 17636, dtype: object
```

8. Most common player position

```
common_position = df['Position'].mode()[0]
print("Most Common Position:", common_position)
```

```
Most Common Position: <span class="pos pos28">SUB
```

9. Find the most frequent nationality among all players

```
most_common_nationality = df['Nationality'].mode()[0]
print("Most Common Nationality:", most_common_nationality)
```

```
Most Common Nationality: England
```

10. List players with an overall rating above 90

```
players_above_90 = df[df['Overall'] > 90][['Name', 'Overall']]
print(players_above_90)
```

```
                Name  Overall
3      K. De Bruyne       91
41   R. Lewandowski       91
56         L. Messi       91
75        K. Mbappé       91
124      K. Benzema       91
```

11. Find the average age of players by nationality (for top 5 nationalities only)

```
top_nationalities = df['Nationality'].value_counts().head(5).index
avg_age_by_nationality = df[df['Nationality'].isin(top_nationalities)].groupby('Nationality')['Age'].mean()
print(avg_age_by_nationality)
```

```
Nationality
Argentina    24.321471
England      22.068583
France       23.446759
Germany      22.714836
Spain        23.917172
Name: Age, dtype: float64
```

12. Find players who have a wage greater than their market value

```
players_wage_greater_value = df[df['Wage'] > df['Value']]
print(players_wage_greater_value[['Name', 'Wage', 'Value']])
```

```
                  Name   Wage    Value
2            M. Acuña   €46K   €46.5M
3        K. De Bruyne  €350K  €107.5M
```

```
5          J. Kimmich  €130K  €105.5M
7         22 Paulinho   €61K   €28.5M
8              E. Can   €63K   €30.5M
...               ...    ...      ...
17652     22 E. Grosz   €500    €180K
17653     22 S. Booth   €850    €110K
17654    22 L. Grimpe   €500    €210K
17655   Deng Xiongtao   €500    €100K
17656  22 Lim Jun Sub   €500    €100K

[10280 rows x 3 columns]
```

13. Find the tallest player and his position

```
df['Height_num'] = df['Height'].str.replace('cm', '').astype(float)
tallest_player = df.loc[df['Height_num'].idxmax()]
print(tallest_player[['Name', 'Height', 'Position']])
```

```
Name                          K. Hudlin
Height                            206cm
Position    <span class="pos pos28">SUB
Name: 12975, dtype: object
```

14. Count how many players are free agents (i.e., Club is NaN)

```
num_free_agents = df['Club'].isnull().sum()
print("Number of Free Agents:", num_free_agents)
```

```
Number of Free Agents: 211
```

15. Identify clubs having players with an overall rating more than 90

```
top_club_players = df[df['Overall'] > 90]['Club'].unique()
print(top_club_players)
```

```
['Manchester City' 'FC Barcelona' 'Paris Saint-Germain' 'Real Madrid CF']
```

16. Find the player with the highest difference between Potential and Overall

```
df['Potential_Overall_Diff'] = df['Potential'] - df['Overall']
biggest_difference_player = df.loc[df['Potential_Overall_Diff'].idxmax()]
print(biggest_difference_player[['Name', 'Potential', 'Overall', 'Potential_Overall_Diff']])
```

```
Name                       D. Lobban
Potential                         79
Overall                           53
Potential_Overall_Diff            26
Name: 12373, dtype: object
```

17. Determine the club having the highest total player wage

```
def wage_to_float(val):
    if isinstance(val, str):
        val = val.replace('€', '')
        if 'K' in val:
            return float(val.replace('K', '')) * 1e3
        elif 'M' in val:
            return float(val.replace('M', '')) * 1e6
        else:
            return float(val)
    return np.nan

df['Wage_num'] = df['Wage'].apply(wage_to_float)
highest_wage_club = df.groupby('Club')['Wage_num'].sum().idxmax()
print("Club with Highest Total Wages:", highest_wage_club)
```

```
Club with Highest Total Wages: Real Madrid CF
```

18. Identify the player with the maximum market value

```
def value_to_float(val):
    if isinstance(val, str):
        val = val.replace('€', '')
        if 'M' in val:
            return float(val.replace('M', '')) * 1e6
        elif 'K' in val:
            return float(val.replace('K', '')) * 1e3
        else:
            return float(val)
    return np.nan

df['Value_num'] = df['Value'].apply(value_to_float)
highest_value_player = df.loc[df['Value_num'].idxmax()]
print(highest_value_player[['Name', 'Value']])
```

```
Name     K. Mbappé
Value      €190.5M
Name: 75, dtype: object
```

19. Find players whose height is above 190 cm and weight is above 85 kg

```
df['Weight_num'] = df['Weight'].str.replace('kg', '').astype(float)

df['Height_num'] = df['Height'].str.replace('cm', '').astype(float)

tall_heavy_players = df[(df['Height_num'] > 190) & (df['Weight_num'] > 85)]
print(tall_heavy_players[['Name', 'Height', 'Weight']])
```

```
                   Name Height Weight
180          E. Haaland  195cm   94kg
192         V. van Dijk  193cm   92kg
259          T. Meunier  191cm   90kg
267        S. McTominay  193cm   88kg
335           T. Souček  192cm   86kg
...                 ...    ...    ...
17581  21 L. Witherbye  192cm   88kg
17606       A. Burnett  192cm   86kg
17608        P. Loretz  193cm   88kg
17614   21 W. McDeeley  192cm   87kg
17639      22 Ş. Dobre  194cm   86kg

[602 rows x 3 columns]
```

20. Find the club with the highest average player age

```
average_age_per_club = df.groupby('Club')['Age'].mean()
club_highest_avg_age = average_age_per_club.idxmax()
highest_avg_age = average_age_per_club.max()

print(f"Club with highest average player age: {club_highest_avg_age} ({highest_avg_age:.2f} years)")
```

```
Club with highest average player age: Barnet (39.00 years)
```