honfinal195150devi-emre-g195188rek

June 16, 2023

1 Football Player Database | Top 5 Leagues

–Yorum(1)–

-> Burada kullanacağımız kütüphaneleri tanıttık. -> numpy: Veri manipülasyonu ve hesaplamalar için kullanılır. -> pandas: Verileri tablo benzeri DataFrame'lerde saklamak için kullanılır. -> seaborn ve matplotlib.pyplot: Bu kütüphaneler ise görselleştirme amaçlı kullanılır. -> xgboost: XGBoost kütüphanesini içe aktarır. XGBoost, Gradient Boosting yöntemini kullanarak güçlü bir makine öğrenimi algoritmasıdır.

[53]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import xgboost
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.model_selection import train_test_split
from sklearn.model_selection import GridSearchCV
```

–Yorum(2)–

-> Buradaki kod pandas kütüphanesini kullanarak bir csv dosyasını DataFrame'ye dönüştürüyor. -> pd.read_csv() fonksiyonu, belirtilen CSV dosyasını okuyarak içeriğini bir DataFrame'e dönüştürür ve df_players adında bir değişkene atar. -> df_players.head() fonksiyonu, oluşturulan DataFrame'in ilk beş satırını ekrana basar. Bu, DataFrame'in yapısını ve verilerin nasıl göründüğünü anlamak için bir önizleme sağlar.

[40]:
```
df_players = pd.read_csv('C:
↪\\Users\\Emre\\Desktop\\archive\\top5_leagues_player.csv', index_col = [0])
df_players.head()
```

[40]:

	name	full_name	age	height	\
0	Ederson	NaN	29	1.88	
1	Stefan Ortega	Stefan Ortega Moreno	30	1.85	
2	Scott Carson	Scott Paul Carson	37	1.88	
3	Rúben Dias	Rúben Santos Gato Alves Dias	26	1.87	
4	Nathan Aké	Nathan Benjamin Aké	28	1.80	

```
                  nationality place_of_birth  price  max_price  \
0            Brazil  Portugal    Osasco (SP)  45.00       70.0
1              Germany  Spain     Hofgeismar   6.00        6.0
2                     England     Whitehaven   0.25        6.0
3                    Portugal        Amadora  75.00       75.0
4  Netherlands  Cote d'Ivoire       Den Haag  35.00       40.0

                  position  shirt_nr   foot      club contract_expires  \
0              Goalkeeper        31   left  Man City       2026-06-30
1              Goalkeeper        18  right  Man City       2025-06-30
2              Goalkeeper        33  right  Man City       2023-06-30
3  Defender - Centre-Back         3  right  Man City       2027-06-30
4  Defender - Centre-Back         6   left  Man City       2025-06-30

  joined_club player_agent outfitter league
0  2017-07-01    Gestifute      Puma    EPL
1  2022-07-01  neblung …       NaN    EPL
2  2021-07-20    Wasserman      Puma    EPL
3  2020-09-29    Gestifute      Nike    EPL
4  2020-08-05    Wasserman      Nike    EPL
```

–Yorum(3)–

-> DataFrame'in sayısal sütunlarının istatistiksel özetini hesaplar. -> .T ifadesi, satırlar ile sütunların yer değiştirmesini sağlar. Bunu yapmaktaki amaç istatistiklerin daha okunur bir hale gelmesini sağlamaktır.

[41]:
```
df_players.describe().T
```

[41]:

	count	mean	std	min	25%	50%	75%	max
age	2612.0	26.305513	4.645297	17.000	23.00	26.00	30.00	42.00
height	2601.0	1.830892	0.067255	1.630	1.78	1.83	1.88	2.06
price	2602.0	10.650711	15.773271	0.025	1.80	4.50	13.00	180.00
max_price	2606.0	17.304729	21.948577	0.025	3.50	10.00	22.00	200.00
shirt_nr	2612.0	20.495789	17.914206	1.000	8.00	17.00	27.00	99.00

–Yorum(4)–

-> Bu kod, df_players DataFrame'inin bir özetini sağlar ve DataFrame'in sütunlarının veri türleri, bellek kullanımı ve eksik değerler hakkında bilgi verir.

[42]:
```
df_players.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2612 entries, 0 to 2611
Data columns (total 17 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
```

```
 0   name              2612 non-null   object
 1   full_name         1480 non-null   object
 2   age               2612 non-null   int64
 3   height            2601 non-null   float64
 4   nationality       2612 non-null   object
 5   place_of_birth    2595 non-null   object
 6   price             2602 non-null   float64
 7   max_price         2606 non-null   float64
 8   position          2612 non-null   object
 9   shirt_nr          2612 non-null   int64
 10  foot              2576 non-null   object
 11  club              2612 non-null   object
 12  contract_expires  2544 non-null   object
 13  joined_club       2612 non-null   object
 14  player_agent      2353 non-null   object
 15  outfitter         1003 non-null   object
 16  league            2612 non-null   object
dtypes: float64(3), int64(2), object(12)
memory usage: 367.3+ KB
```

–Yorum(5)–

-> Burada eksik değerlerin olduğu sütunları tespit edip bunları yeni bir değişkene atıyorum. -> Bu kodun amacı, df_players DataFrame'inin içinde eksik değerler içeren sütunları tespit etmek, bu sütunlara ait eksik değerleri içeren bir DataFrame oluşturmak ve eksik değerlerin sayısını hesaplamaktır. Bu bilgiler, veri setindeki eksik değerlerin durumunu değerlendirmek ve ilgili işlemleri yapmak için kullanılabilir.

[43]:
```
empty_cols = df_players.columns[df_players.isna().any()].tolist()

df_isnull = df_players[empty_cols]

print(df_isnull.isnull().sum())
```

```
full_name          1132
height               11
place_of_birth       17
price                10
max_price             6
foot                 36
contract_expires     68
player_agent        259
outfitter          1609
dtype: int64
```

–Yorum(6)–

-> Ardından veri setinde eksik değerleri olan sütunları çıkartıp kalan satır bigisini ekrana yazdırıyorum. -> Bu kod, df_players DataFrame'inden belirli sütunlarda eksik değeri olan satırları çıkarır ve DataFrame'in yeni satır ve sütun sayısını döndürür. -> subset kodu ile boş değeri olan sütunları

seçiyoruz ve ilgili satırı DataFrame'den siliyoruz. -> df_players.shape kodu ile DataFrame'in ilgili sütunların eksik satırları çıkarıldıktan sonra kalan satır ve sütun sayısını ekrana yazdırıyoruz.

[44]:
```
df_players = df_players.dropna(subset=['contract_expires', 'height', 'foot',␣
↪'price', 'max_price'])
df_players.shape
```

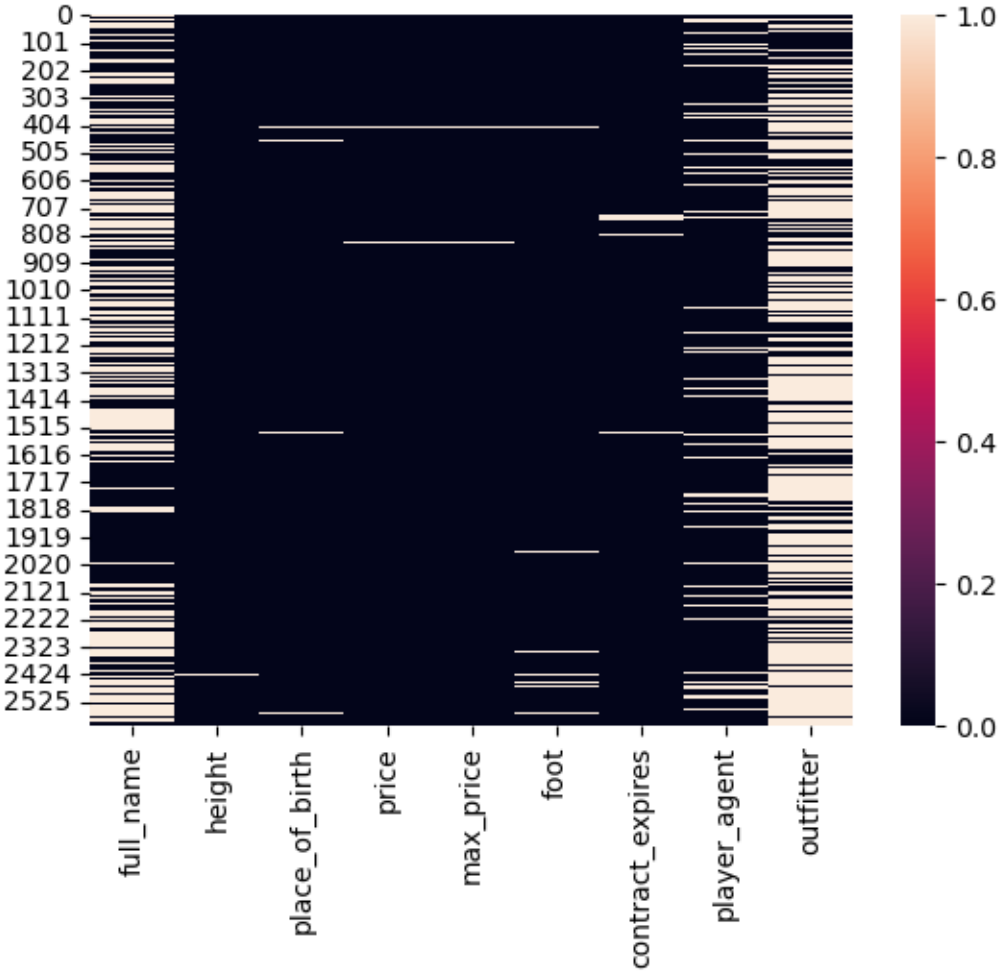
[44]:
```
(2498, 17)
```

–Yorum(7)–

-> Bu kod, eksik değerlerimizin olduğu veri setindeki değerleri görselleştirmek için bir ısı haritası oluşturur. -> df_isnull.isnull() DataFrame'inin üzerine bir ısı haritası oluşturulur. İkinci isnull() fonksiyonu, her eksik değer için True değerini 1 olarak temsil eden bir matris döndürür ve bu matris ısı haritası olarak görselleştirilir.

[45]:
```
sns.heatmap(df_isnull.isnull())
```

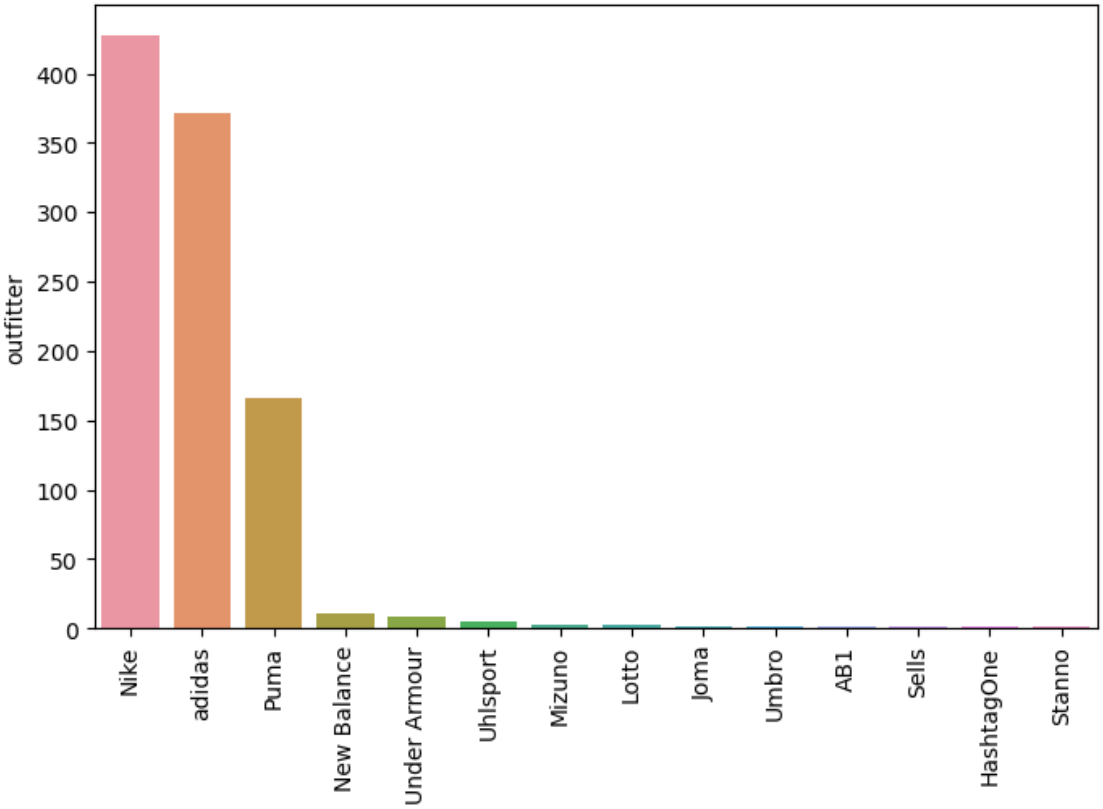
[45]:
```
<Axes: >
```



–Yorum(8)–

-> Bu kod, eksik değerleri içeren DataFrame'deki 'outfitter' değerlerinin sayısını hesaplar ve bu değerleri bir çubuk grafikle görselleştirir. Ayrıca, x-ekseni etiketlerini 90 derece döndürür. -> plt.figure(figsize=(8,5)) kodu yeni bir çizim alanı oluşturur ve bu alanın genişlik ve yüksekliğini belirtir.

[48]:
```
plt.figure(figsize=(8,5))

outfitter_counts = df_isnull['outfitter'].value_counts()
sns.barplot(data=df_isnull, x=outfitter_counts.index, y=outfitter_counts)

ax = plt.gca()
ax.tick_params(axis='x', labelrotation=90)
```

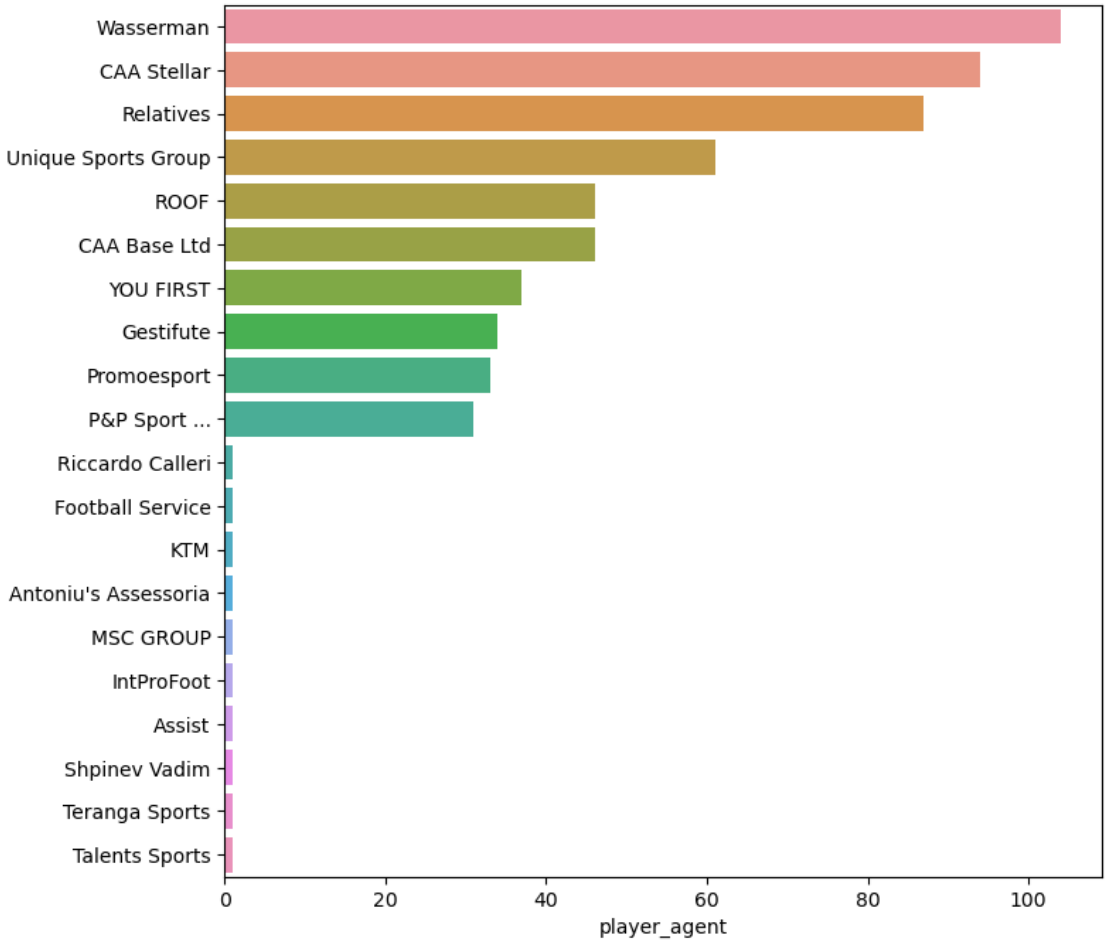


–Yorum(9)–

-> Bu kod, eksik değerleri içeren DataFrame'deki 'player_agent' sütunundaki farklı değerlerin sayısını hesaplar. Ardından, en çok ve en az görülen 10 değeri seçer ve bu değerleri çubuk grafikle görselleştirir.

[49]:
```
plt.figure(figsize=(8,8))

outfitter_counts = df_isnull['player_agent'].value_counts()

top_10 = outfitter_counts.head(10)
last_10 = outfitter_counts.tail(10)
plot_data = pd.concat([top_10, last_10])

sns.barplot(data=df_isnull, x=plot_data, y=plot_data.index)
```

[49]:
```
<Axes: xlabel='player_agent'>
```



–Yorum(10)–

-> Bu kodun amacı, DataFrame içerisindeki boş değerleri belirli bir yerine ("unknown" olarak) değiştirmektir. Böylece, eksik verilerin yerine geçici bir değer atanmış olur.

[50]:
```
df_players = df_players.replace(np.nan, 'unknown')
```

[51]:
```
df_players.head()
```

[51]:

	name	full_name	age	height	\
0	Ederson	unknown	29	1.88	
1	Stefan Ortega	Stefan Ortega Moreno	30	1.85	
2	Scott Carson	Scott Paul Carson	37	1.88	
3	Rúben Dias	Rúben Santos Gato Alves Dias	26	1.87	
4	Nathan Aké	Nathan Benjamin Aké	28	1.80	

	nationality	place_of_birth	price	max_price	\
0	Brazil	Portugal	Osasco (SP)	45.00	70.0
1	Germany	Spain	Hofgeismar	6.00	6.0
2	England		Whitehaven	0.25	6.0
3	Portugal		Amadora	75.00	75.0
4	Netherlands	Cote d'Ivoire	Den Haag	35.00	40.0

	position	shirt_nr	foot	club	contract_expires	\
0	Goalkeeper	31	left	Man City	2026-06-30	
1	Goalkeeper	18	right	Man City	2025-06-30	
2	Goalkeeper	33	right	Man City	2023-06-30	
3	Defender - Centre-Back	3	right	Man City	2027-06-30	
4	Defender - Centre-Back	6	left	Man City	2025-06-30	

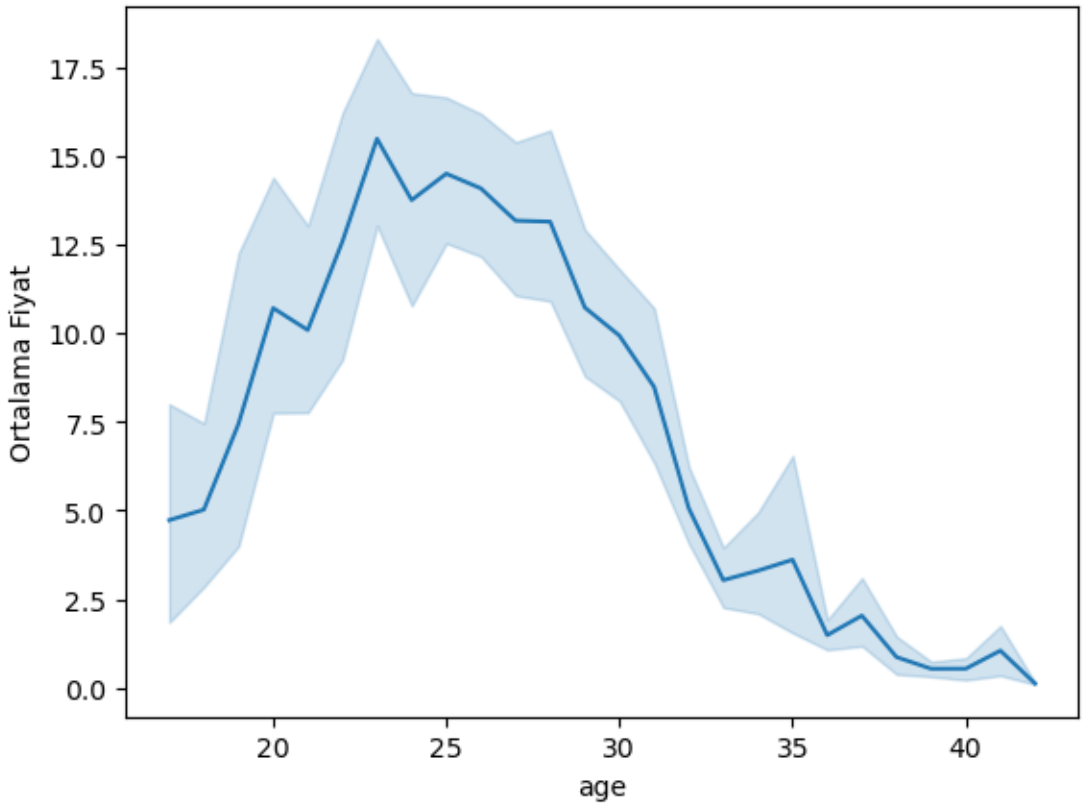
	joined_club	player_agent	outfitter	league
0	2017-07-01	Gestifute	Puma	EPL
1	2022-07-01	neblung …	unknown	EPL
2	2021-07-20	Wasserman	Puma	EPL
3	2020-09-29	Gestifute	Nike	EPL
4	2020-08-05	Wasserman	Nike	EPL

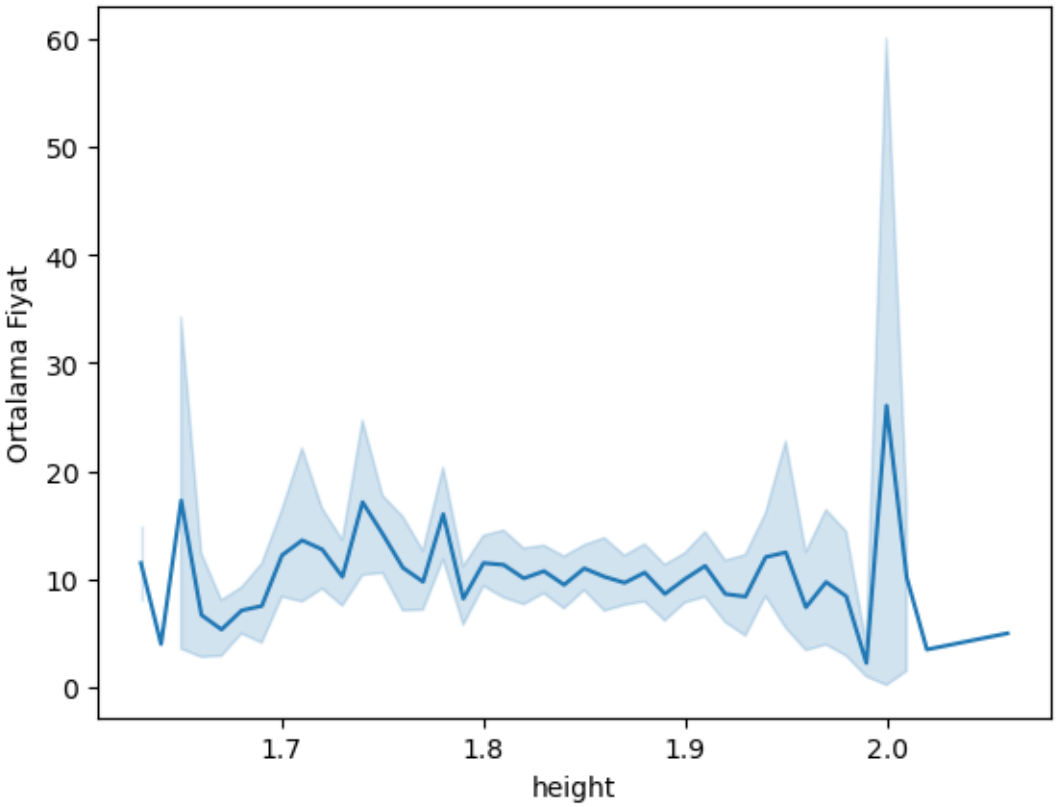
–Yorum(11)–

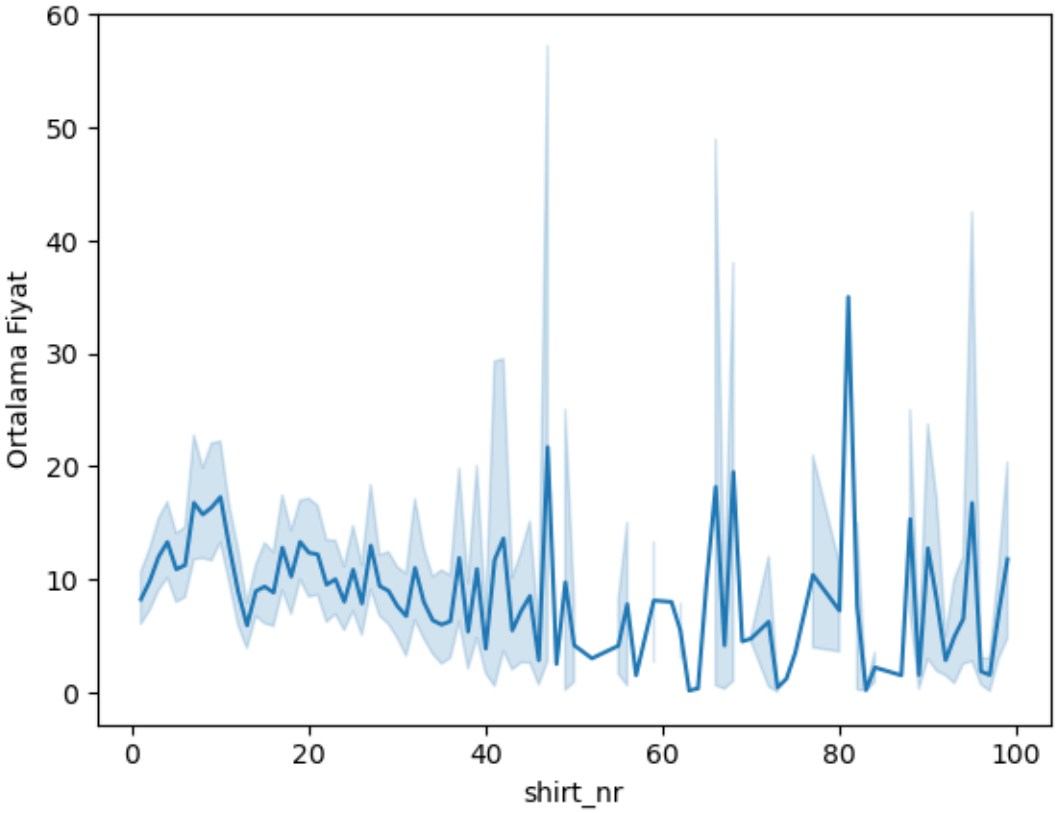
-> Bu kodun amacı, verilen DataFrame üzerinde belirli bir özellik (feature) bazında grafikler oluşturmak ve görsel analiz yapmaktır.

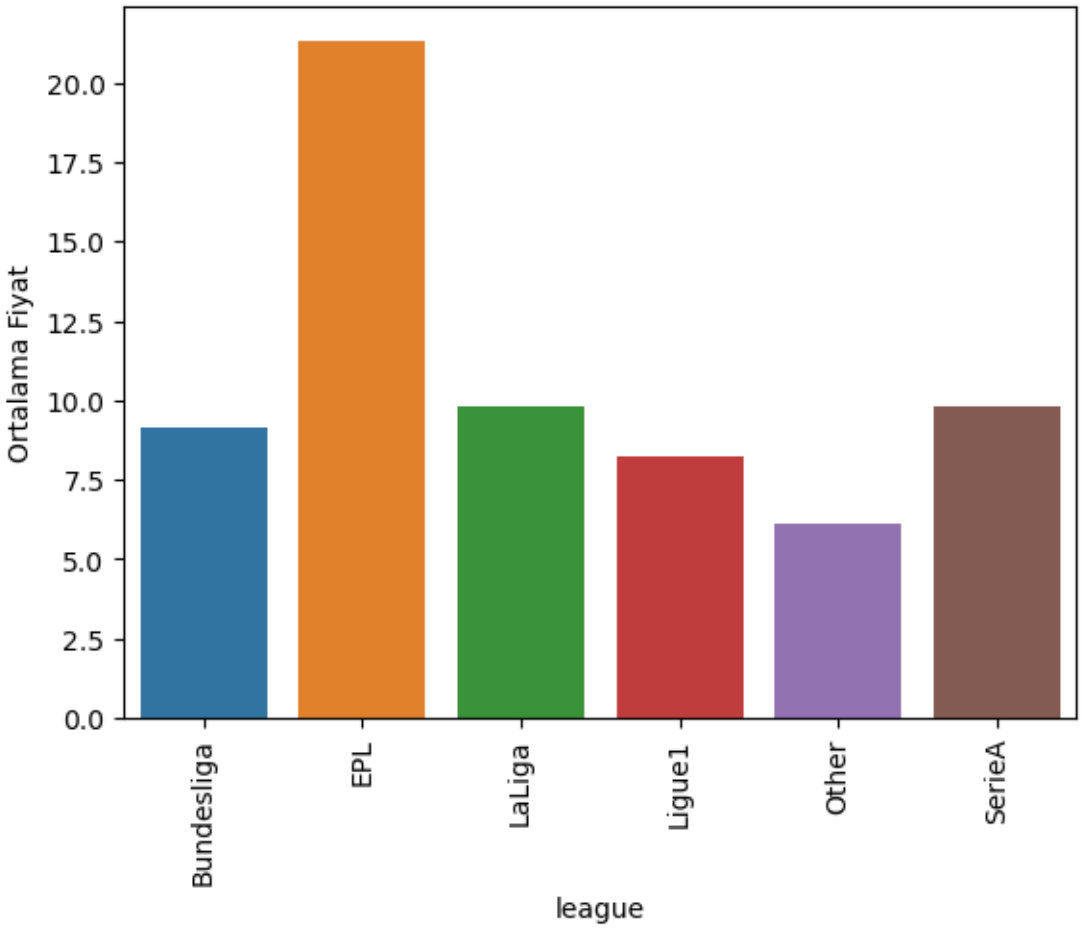
[52]:
```
def plotting(df, feature):
    df_mean = df.groupby(feature)['price'].mean().reset_index()
    sns.barplot(data=df_mean, x=feature, y='price')

    ax = plt.gca()
    ax.tick_params(axis='x', labelrotation=90)
    plt.ylabel('Ortalama Fiyat')
    plt.show()

def plottingLine(df, feature):
    sns.lineplot(data=df, x=feature, y='price')
    plt.ylabel('Ortalama Fiyat')
    plt.show()
```

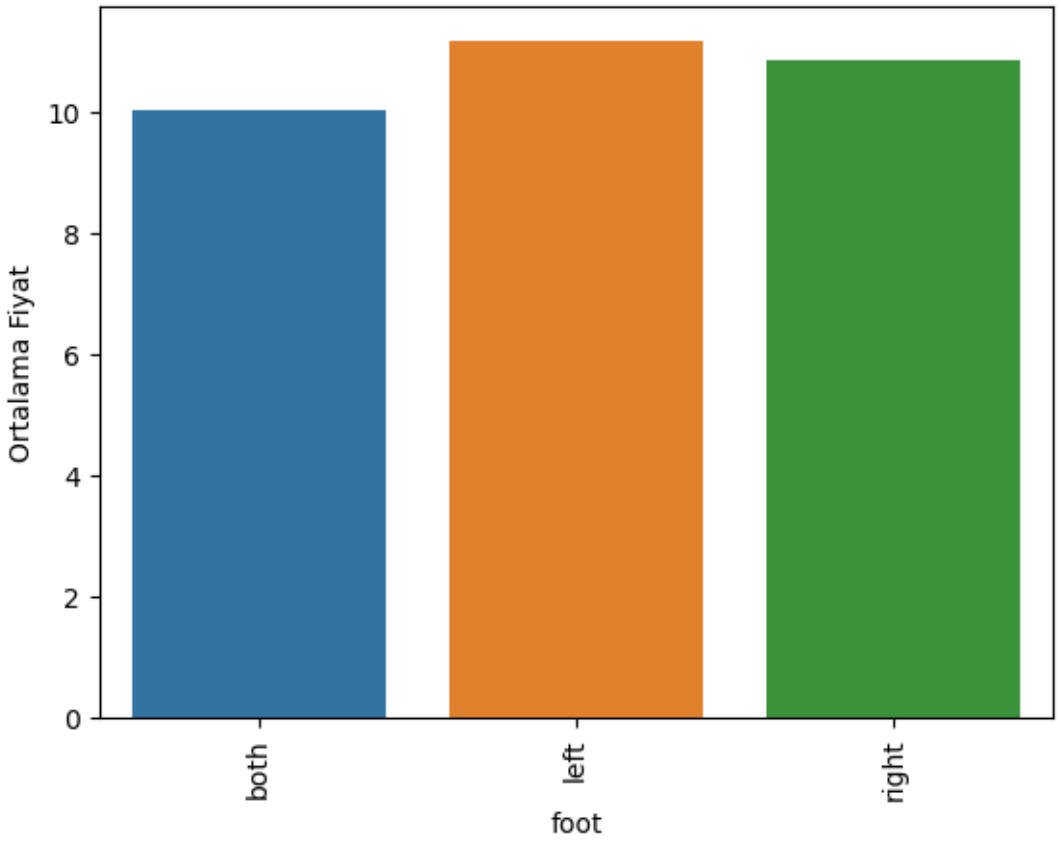
```
feature_list = ['league', 'foot', 'position', 'contract_expires', 'outfitter']

numarical_features = ['age', 'height', 'shirt_nr']

for feature in numarical_features:
    plottingLine(df_players, feature)

for feature in feature_list:
    plotting(df_players, feature)
```

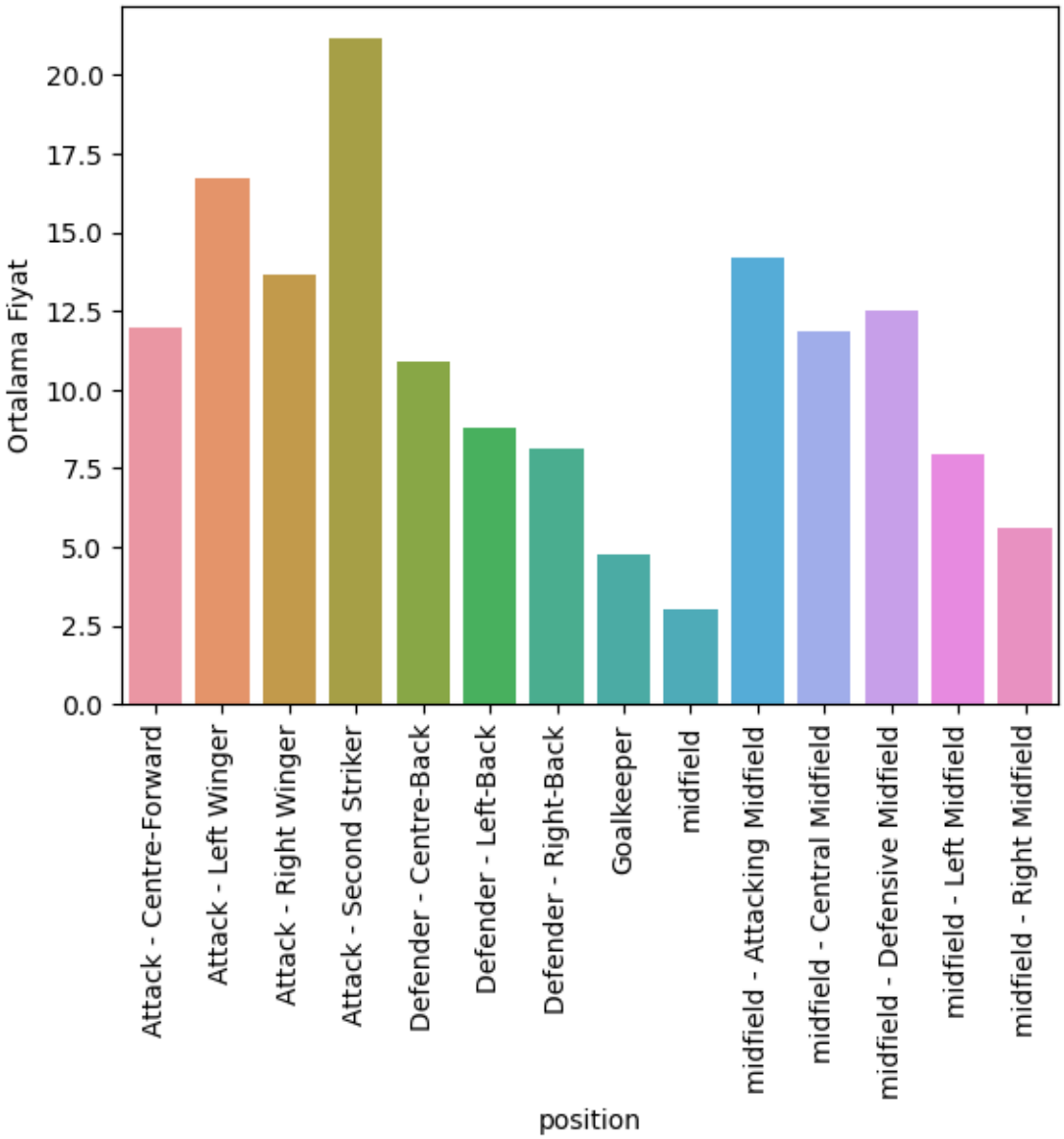


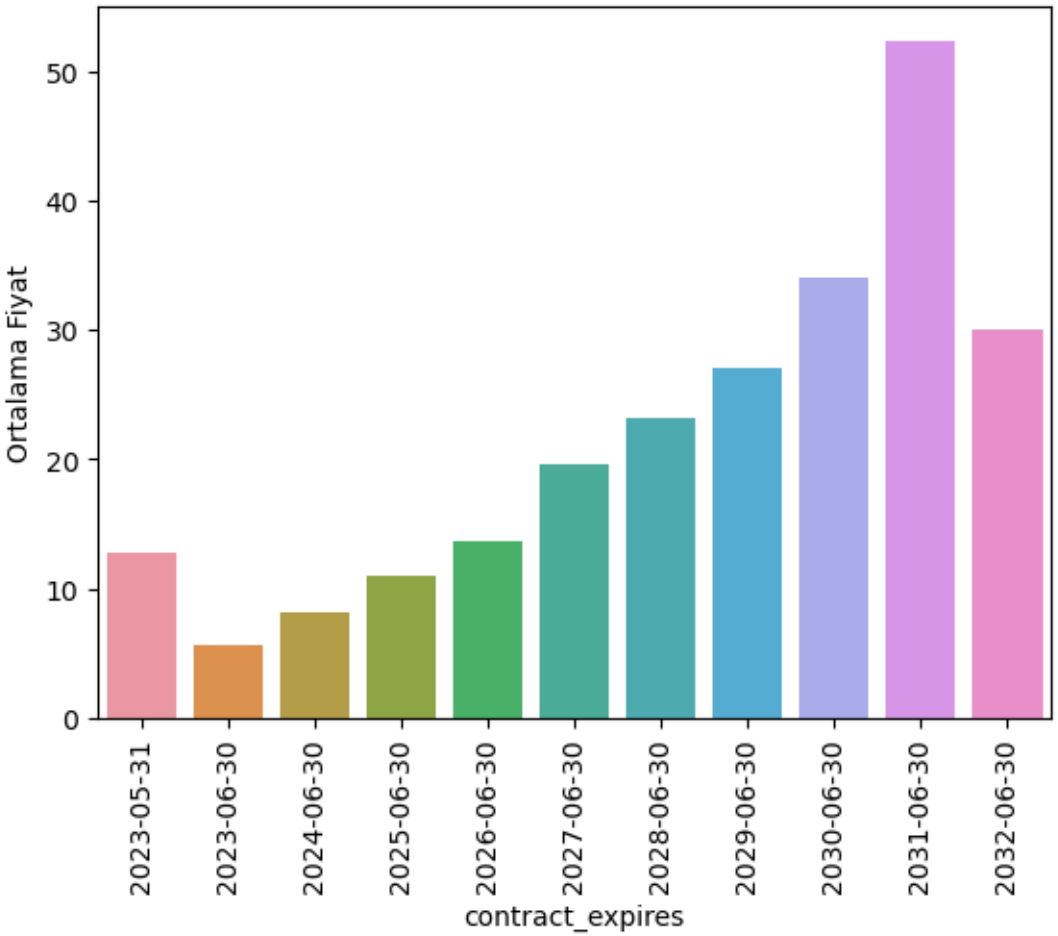

60
50
40
30
20
10
0
Ortalama Fiyat
1.7
1.8
1.9
2.0
height

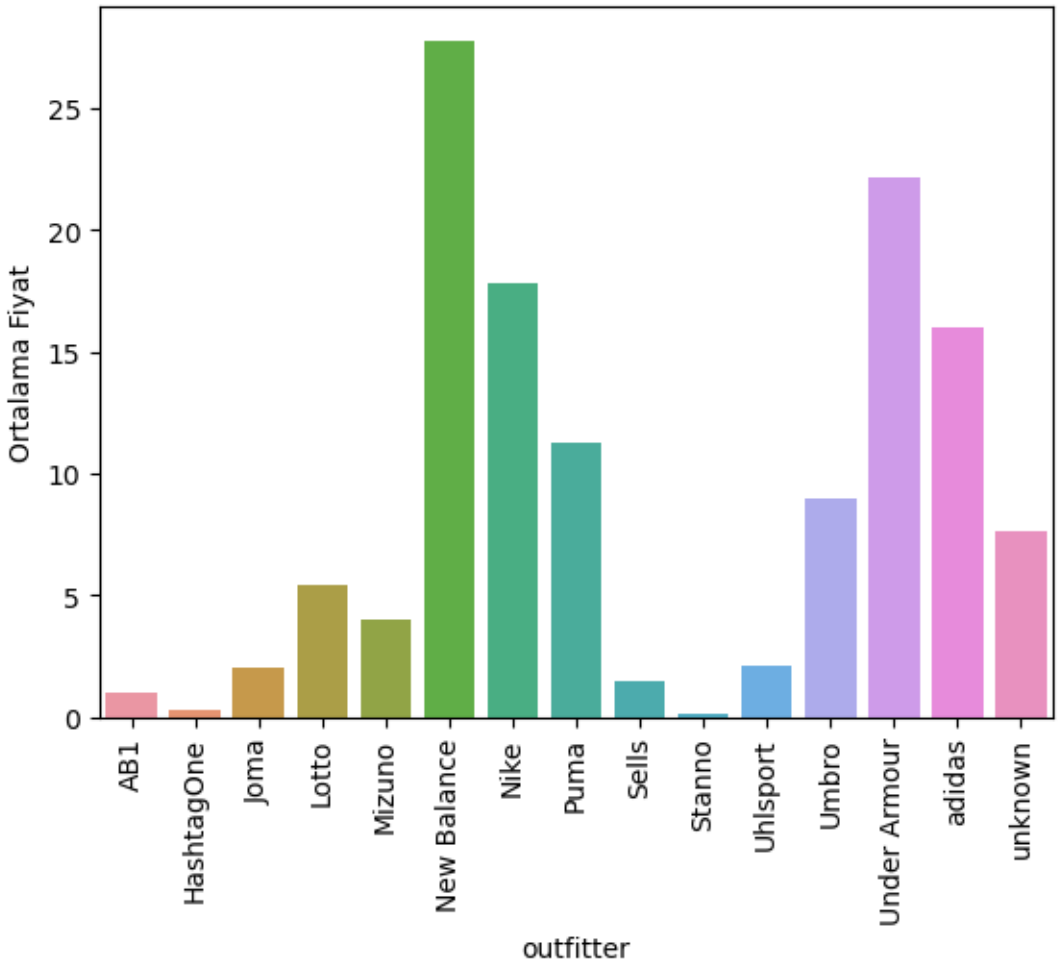

Ortalama Fiyat
shirt_nr
0
10
20
30
40
50
60
0
20
40
60
80
100


Ortalama Fiyat
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
Bundesliga
EPL
LaLiga
Ligue1
Other
SerieA
league


Ortalama Fiyat
0
2
4
6
8
10
both
left
right
foot


Ortalama Fiyat
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
Attack - Centre-Forward
Attack - Left Winger
Attack - Right Winger
Attack - Second Striker
Defender - Centre-Back
Defender - Left-Back
Defender - Right-Back
Goalkeeper
midfield
midfield - Attacking Midfield
midfield - Central Midfield
midfield - Defensive Midfield
midfield - Left Midfield
midfield - Right Midfield
position


Ortalama Fiyat
0
10
20
30
40
50
2023-05-31
2023-06-30
2024-06-30
2025-06-30
2026-06-30
2027-06-30
2028-06-30
2029-06-30
2030-06-30
2031-06-30
2032-06-30
contract_expires

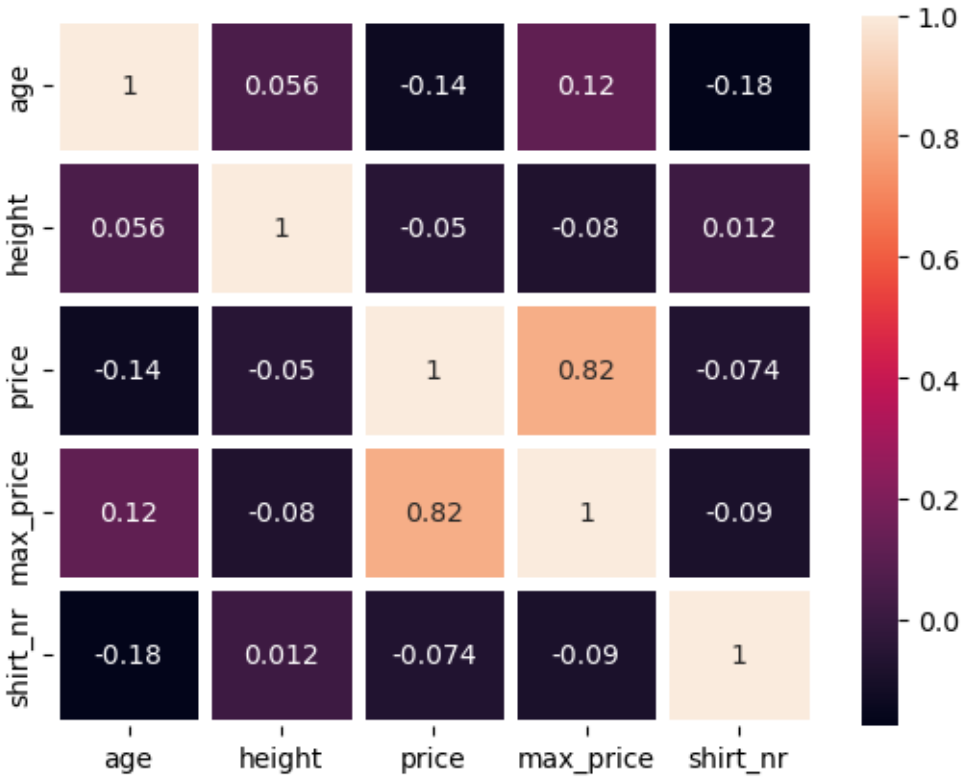


–Yorum(12)–

-> Bu kodun amacı, DataFrame içerisindeki sayısal sütunlar arasındaki korelasyonları hesaplamak ve bu korelasyonları görsel olarak anlaşılabilir bir şekilde ısı haritası olarak göstermektir. Böylece, veri setindeki değişkenler arasındaki ilişkiler ve güçlü korelasyonlar daha anlaşılır olur.

[34]:
```
sns.heatmap(df_players.corr(numeric_only=True), annot=True, linewidth=5)
```

[34]: <Axes: >



–Yorum(13)–

-> Bu kodun amacı, veri setini hedef değişken ve özellikler olmak üzere iki ayrı DataFrame'e bölmektir. -> Bu bölme işlemi, veri setinin modelin eğitimi ve testi için kullanılacak bağımsız değişkenlerini ve hedef değişkenini ayırmak için yapılmaktadır. Model, df_features DataFrame'indeki özellikleri kullanarak df_target DataFrame'indeki hedef değişkeni tahmin etmeye çalışacaktır.

[20]:
```
df_target = df_players[['price']]
df_features = df_players[['age', 'height', 'league','foot', 'position', 'club',
                          'contract_expires', 'joined_club', 'player_agent',␣
 ↪'outfitter', 'nationality']]
```

–Yorum(14)–

-> Bu kodun amacı, df_features DataFrame'inin her bir sütununda bulunan benzersiz değerleri belirlemek ve bu değerleri sütun adıyla birlikte ekrana yazdırmaktır. Böylece, her bir özellik sütununda hangi benzersiz değerlerin bulunduğunu görebilirsiniz.

[21]:
```
for column in df_features.columns:
    unique_values = df_features[column].unique()
    print(f"Unique values in column '{column}': {unique_values}")
```

```
Unique values in column 'age': [29 30 37 26 28 27 22 32 18 20 31 21 23 19 38 24
33 25 36 35 34 41 39 17
```

```
 40 42]
Unique values in column 'height': [1.88 1.85 1.87 1.8  1.89 1.71 1.83 1.69 1.91
1.79 1.77 1.81 1.73 1.82
 1.95 1.7  1.94 1.86 1.93 1.92 1.78 1.9  1.75 1.72 1.74 1.96 1.84 1.97
 2.01 1.98 1.63 1.76 1.67 1.99 1.65 2.   1.68 1.66 2.02 1.64 2.06]
Unique values in column 'league': ['EPL' 'Other' 'Bundesliga' 'SerieA' 'LaLiga'
'Ligue1']
Unique values in column 'foot': ['left' 'right' 'both']
Unique values in column 'position': ['Goalkeeper' 'Defender - Centre-Back'
'Defender - Left-Back'
 'Defender - Right-Back' 'midfield - Defensive Midfield'
 'midfield - Central Midfield' 'midfield - Attacking Midfield'
 'Attack - Left Winger' 'Attack - Right Winger' 'Attack - Centre-Forward'
 'Attack - Second Striker' 'midfield - Left Midfield'
 'midfield - Right Midfield' 'midfield']
Unique values in column 'club': ['Man City' 'Chelsea' 'Chelsea U21' 'Arsenal'
'Liverpool' 'Man Utd'
 'Tottenham' 'Newcastle' 'West Ham' 'Leicester' 'Aston Villa' 'Wolves'
 'Southampton' 'Brighton' 'Everton' 'Nottm Forest' 'Brentford' 'Leeds'
 'Crystal Palace' 'Fulham' 'Bournemouth' 'Bayern Munich' 'Bor. Dortmund'
 'B. Dortmund II' 'RB Leipzig' 'RB Leipzig U19' 'RB Leipzig U17'
 'B. Leverkusen' 'E. Frankfurt' "Bor. M'gladbach" 'VfL Wolfsburg'
 'SC Freiburg' 'TSG Hoffenheim' 'Union Berlin' 'FC Augsburg'
 'VfB Stuttgart' '1.FSV Mainz 05' '1.FC Köln' '1.FC Köln II' 'Hertha BSC'
 'Hertha BSC II' 'Hertha BSC U19' 'Werder Bremen' 'FC Schalke 04'
 'VfL Bochum' 'SSC Napoli' 'AC Milan' 'Inter' 'Juventus' 'Atalanta BC'
 'AS Roma' 'Lazio' 'Fiorentina' 'Sassuolo' 'Torino' 'Udinese Calcio'
 'Bologna' 'Monza' 'FC Empoli' 'Salernitana' 'Lecce' 'Spezia Calcio'
 'Hellas Verona' 'Sampdoria' 'Cremonese' 'Real Madrid' 'RM Castilla'
 'Barcelona' 'Barça Atlètic' 'Atlético Madrid' 'Real Sociedad'
 'Villarreal' 'Real Betis' 'Betis Deportivo' 'Athletic' 'Valencia'
 'Valencia B' 'Sevilla FC' 'Celta de Vigo' 'Celta Vigo B' 'Getafe'
 'CA Osasuna' 'CA Osasuna Prom' 'Girona' 'Espanyol' 'RCD Espanyol B'
 'Rayo Vallecano' 'RCD Mallorca' 'UD Almería' 'Real Valladolid' 'Cádiz CF'
 'Elche CF' 'Paris SG' 'Monaco' 'Monaco U21' 'Rennes' 'Marseille'
 'OGC Nice' 'Olympique Lyon' 'LOSC Lille' 'Lens' 'FC Nantes' 'Montpellier'
 'R. Strasbourg' 'Stade Reims' 'FC Lorient' 'Toulouse' 'Troyes'
 'Stade Brestois' 'Angers SCO' 'AJ Auxerre' 'Clermont Foot' 'AC Ajaccio']
Unique values in column 'contract_expires': ['2026-06-30' '2025-06-30'
'2023-06-30' '2027-06-30' '2024-06-30'
 '2028-06-30' '2029-06-30' '2030-06-30' '2031-06-30' '2023-05-31'
 '2032-06-30']
Unique values in column 'joined_club': ['2017-07-01' '2022-07-01' '2021-07-20'
'2020-09-29' '2020-08-05'
 '2016-08-09' '2018-01-30' '2022-09-01' '2022-08-16' '2017-07-14'
 '2019-07-04' '2022-07-04' '2023-01-23' '2016-07-01' '2015-08-30'
 '2021-07-01' '2021-08-05' '2018-07-10' '2022-01-31' '2020-09-24'
 '2018-08-08' '2022-08-02' '2021-07-28' '2022-08-31' '2023-01-05'
```

```
'2022-07-16' '2020-08-28' '2022-08-05' '2020-08-26' '2022-01-01'
'2019-07-01' '2012-08-24' '2023-01-31' '2016-07-16' '2015-01-01'
'2022-08-04' '2020-09-04' '2023-01-15' '2022-07-13' '2019-01-02'
'2023-01-20' '2020-07-01' '2023-01-11' '2023-01-01' '2021-08-20'
'2019-07-25' '2020-09-01' '2016-07-22' '2022-07-22' '2019-08-08'
'2021-07-30' '2021-08-31' '2020-10-05' '2016-01-14' '2020-08-01'
'2019-07-02' '2019-01-01' '2018-07-19' '2019-08-05' '2018-01-01'
'2015-07-01' '2017-07-21' '2020-08-10' '2018-07-01' '2020-09-18'
'2011-07-01' '2017-08-31' '2022-01-30' '2020-09-19' '2023-01-06'
'2021-07-02' '2022-07-27' '2021-08-14' '2014-07-01' '2022-07-05'
'2022-08-22' '2022-07-15' '2020-09-02' '2020-01-29' '2016-01-01'
'2021-07-23' '2022-08-30' '2015-09-01' '2023-01-13' '2012-08-31'
'2021-08-01' '2014-07-31' '2017-08-23' '2022-07-08' '2020-01-01'
'2014-07-23' '2020-08-11' '2021-08-27' '2015-08-28' '2023-01-25'
'2018-01-31' '2011-01-01' '2022-09-12' '2020-07-03' '2018-07-26'
'2014-08-09' '2020-09-08' '2012-07-01' '2022-01-07' '2021-08-13'
'2019-07-23' '2019-08-02' '2020-09-07' '2019-01-31' '2017-07-19'
'2022-08-26' '2021-08-28' '2022-08-17' '2015-07-10' '2022-08-23'
'2014-07-03' '2020-10-02' '2020-07-27' '2022-07-07' '2022-08-29'
'2021-01-29' '2020-01-31' '2022-07-26' '2018-07-20' '2022-08-12'
'2018-08-09' '2016-01-22' '2021-07-15' '2020-09-03' '2017-01-03'
'2016-07-03' '2019-07-08' '2023-01-29' '2017-08-03' '2020-09-16'
'2013-07-01' '2019-07-11' '2022-01-27' '2022-01-13' '2021-08-04'
'2020-09-09' '2022-07-12' '2023-01-22' '2021-07-09' '2020-09-23'
'2017-07-08' '2022-08-18' '2023-01-30' '2018-07-24' '2020-01-30'
'2022-01-17' '2023-01-17' '2016-08-01' '2021-12-06' '2020-08-12'
'2021-08-25' '2021-08-02' '2023-01-26' '2022-07-06' '2020-10-04'
'2022-07-09' '2021-08-10' '2022-07-25' '2022-07-29' '2019-08-03'
'2010-07-01' '2020-09-10' '2020-07-29' '2021-02-01' '2019-01-24'
'2020-10-18' '2022-07-02' '2022-08-08' '2015-08-13' '2017-07-03'
'2022-01-04' '2009-01-01' '2022-08-09' '2017-01-01' '2021-07-22'
'2022-07-28' '2016-08-31' '2022-07-10' '2022-07-20' '2022-07-03'
'2022-09-07' '2023-01-16' '2022-08-13' '2022-07-31' '2022-08-14'
'2021-02-02' '2022-08-19' '2022-07-21' '2023-02-02' '2020-08-15'
'2019-07-06' '2022-07-14' '2021-07-21' '2021-09-09' '2019-08-07'
'2020-10-03' '2019-07-10' '2018-08-21' '2021-01-14' '2022-08-10'
'2023-01-04' '2021-07-31' '2020-08-29' '2023-01-03' '2014-08-13'
'2021-07-06' '2016-08-11' '2015-08-04' '2018-01-18' '2023-01-14'
'2018-07-31' '2021-07-18' '2016-07-05' '2020-10-14' '2022-07-11'
'2017-07-24' '2017-01-31' '2014-09-01' '2017-01-13' '2015-02-02'
'2021-07-08' '2015-08-20' '2020-08-20' '2020-08-30' '2020-01-24'
'2016-08-26' '2021-07-24' '2018-07-30' '2022-08-07' '2019-01-22'
'2014-01-28' '2018-08-07' '2019-07-29' '2016-07-08' '2022-08-01'
'2023-01-19' '2019-01-04' '2023-01-27' '2022-07-19' '2021-07-05'
'2015-07-02' '2020-07-15' '2009-07-01' '2018-07-02' '2016-01-04'
'2020-07-23' '2020-01-12' '2021-07-27' '2022-10-09' '2015-08-31'
'2014-08-22' '2020-01-25' '2021-01-01' '2015-01-07' '2019-07-18'
'2013-07-04' '2021-07-10' '2021-08-16' '2015-07-16' '2021-07-12'
```

```
'2021-01-27' '2021-01-13' '2019-07-30' '2021-08-26' '2021-07-19'
'2018-07-05' '2019-07-27' '2022-08-21' '2021-07-11' '2022-01-20'
'2020-07-13' '2022-01-11' '2019-08-14' '2021-08-06' '2021-01-21'
'2018-07-13' '2010-01-01' '2019-07-22' '2015-01-21' '2020-08-25'
'2021-07-16' '2020-10-01' '2012-01-01' '2022-01-28' '2021-08-22'
'2020-04-16' '2021-08-18' '2018-08-31' '2022-07-23' '2022-08-03'
'2023-01-18' '2019-01-15' '2023-01-02' '2020-01-07' '2016-01-15'
'2020-08-04' '2019-08-06' '2018-10-24' '2016-01-09' '2021-09-07'
'2020-07-28' '2022-09-19' '2020-07-24' '2019-08-29' '2019-08-13'
'2015-11-30' '2022-11-18' '2023-01-24' '2016-01-05' '2022-03-08'
'2022-01-26' '2022-02-04' '2019-01-18' '2020-01-17' '2020-09-21'
'2020-09-15' '2014-01-01' '2022-01-25' '2015-08-19' '2020-01-27'
'2019-08-01' '2021-08-09' '2022-07-30' '2016-02-01' '2021-10-07'
'2019-01-03' '2017-08-04' '2021-09-28' '2022-10-21' '2023-01-07'
'2022-08-11' '2020-08-14' '2019-08-27' '2020-08-24' '2019-07-05'
'2021-08-19' '2020-01-20' '2018-01-03' '2020-01-15' '2020-01-11'
'2016-08-04' '2019-07-24' '2019-08-23' '2022-08-20' '2020-09-11'
'2021-10-13' '2019-08-04' '2021-08-30' '2020-09-12' '2021-07-17'
'2020-01-02' '2012-07-09' '2017-07-07' '2017-08-30' '2014-01-30'
'2017-08-09' '2017-08-01' '2018-07-04' '2018-08-02' '2015-02-01'
'2017-07-20' '2023-04-01' '2022-12-19' '2021-08-17' '2012-06-22'
'2009-08-01' '2016-01-18' '2015-08-26' '2021-01-04' '2021-08-24'
'2017-08-10' '2022-08-27' '2021-07-13' '2022-11-01' '2022-10-01'
'2021-01-30' '2022-08-28' '2008-07-01' '2019-07-12' '2015-08-06'
'2016-07-27' '2019-07-07' '2023-01-28' '2022-01-03' '2022-01-29'
'2019-07-26' '2013-09-02' '2016-08-30' '2021-08-11' '2020-09-17'
'2017-07-13' '2019-08-20' '2021-01-31' '2022-07-18' '2023-03-07'
'2019-08-12' '2020-09-28' '2019-08-30' '2019-07-13' '2020-09-22'
'2022-07-17' '2013-01-31' '2022-08-25' '2021-07-29' '2021-01-16'
'2021-08-12' '2023-01-21' '2019-08-09' '2023-01-12' '2020-09-05'
'2020-10-07' '2018-12-14' '2019-07-20' '2020-01-14' '2023-02-17'
'2022-01-10' '2023-02-10' '2019-01-11' '2022-10-28' '2019-08-17'
'2020-08-27' '2012-09-01' '2015-07-31' '2014-07-17' '2012-08-27'
'2018-07-12' '2009-07-09' '2018-08-29' '2019-09-02' '2017-08-25'
'2014-07-16' '2013-04-25' '2015-07-20' '2020-11-03' '2021-07-26'
'2020-08-17' '2021-01-24' '2019-07-09' '2018-11-01' '2020-07-21'
'2020-08-31' '2018-07-03' '2021-07-04' '2022-09-28' '2017-08-18'
'2020-09-14' '2019-07-15' '2019-07-03' '2021-01-26' '2022-01-14'
'2020-01-16' '2018-08-01' '2021-01-23' '2015-07-07' '2022-08-06'
'2015-07-08' '2018-07-16' '2017-07-17' '2019-08-19' '2016-07-07'
'2021-04-01' '2021-09-03' '2022-08-24' '2021-07-14' '2018-07-27'
'2018-08-23' '2020-01-23' '2022-09-13' '2021-09-04' '2021-07-07'
'2019-09-01' '2022-03-03' '2017-07-04' '2017-07-05' '2022-03-07'
'2022-01-21' '2019-01-25' '2019-01-28' '2016-08-16' '2017-08-07'
'2019-01-05' '2021-09-01' '2020-10-09' '2021-02-04' '2021-08-23'
'2020-09-20' '2023-02-14' '2022-08-15' '2020-09-25' '2013-07-19'
'2012-07-18' '2020-08-07' '2019-08-24' '2022-09-21' '2020-07-06'
'2019-10-11' '2021-07-03' '2022-01-18' '2019-09-03' '2023-01-09'
```

```
 '2017-01-30' '2016-08-23' '2018-07-17' '2020-08-03' '2022-09-04'
 '2017-07-26' '2018-07-15' '2019-07-17' '2020-08-09' '2020-07-02'
 '2020-04-19' '2022-11-15' '2020-08-18' '2017-08-12' '2017-07-27'
 '2022-09-22' '2017-09-06' '2019-07-16' '2017-07-18' '2019-09-27'
 '2019-09-25' '2020-10-19' '2018-08-30' '2023-03-01' '2019-01-26'
 '2022-09-05' '2022-02-02' '2021-08-29' '2023-02-01' '2020-07-08'
 '2022-01-06' '2021-03-17' '2022-12-09' '2018-04-16' '2018-08-13'
 '2021-01-22' '2021-10-12' '2021-01-05' '2022-10-12' '2021-02-16'
 '2021-01-15' '2022-09-20']
Unique values in column 'player_agent': ['Gestifute' 'neblung …' 'Wasserman'
'unknowen' 'IFM' 'SEG'
 'CAA Base Ltd' 'Pablo Barquero' 'CAA Stellar' 'KIN Partners'
 'Rafaela Pimenta' 'Fernando Hidalgo' 'LIAN Sports Group' 'AC Talent'
 'SXI' 'World in Motion' 'TTMA' 'Paulo Tonietto' 'EMG Mundial'
 'Unique Sports Group' 'Relatives' 'DE 9 FÚTBOL' 'KDS FM' 'HCM Sports …'
 'VPA Sports' 'Elite Management …' 'ROOF' 'ALIK FOOTBALL …'
 'HZ INTERNATIONAL' 'FA Sport' 'Aneke/PMG' 'Premier Sports …'
 'ND SPORTS MANAGEMENT' 'Bertolucci Sports' 'FairSport' 'Anatoliy Patuk'
 'UDN SPORTS' 'JJ SPORT' 'Brezilya Agency' 'Signature - ISCM AG'
 'Nordic Sky' 'Jordan Wise' 'TFM Agency' 'JC …'
 'Elite Project Group …' 'Energy Sports' 'NWS - Neis World …'
 'Bahía Internacional' 'ASBW Sport Marketing' 'PLG' 'Prosport …'
 'F.J.VILLAVERDE' 'The Player …' 'P&P Sport …' 'Colossal Sports …'
 'Concilium Sport' 'Nomi Sports' 'ROGON' 'Triple S Sports'
 '10Ten Football' 'Score Futbol' 'CLOSEVIEW' 'Proeleven S.A.'
 'Best of You' 'The Talents …' 'Fabio Mello Sports' 'MRP.POSITIONUMBER'
 '#LEADERS' '4ComM' 'USFA Management' 'One II Agency'
 'Remington Ellis …' 'Ciro Palermo' 'Villarreal & Schena' 'RDF Football'
 'ARETÉ' 'Gol International' 'Back Sports S.A.' 'AMS CONSULTING'
 'Panthera Sports' 'World Soccer Agency' 'CK66' 'SPOCS Global Sports'
 'SMI Sports …' 'Muy Manero' 'Sports360 GmbH' 'Straight Ace Sports'
 'Global Soccer …' 'Osvaldo Daniel Campo' 'New Era Global …'
 'Universal Twenty Two' 'Trivela SM' 'Dirk Hebel'
 'Branchini Associati …' 'PS Sports' 'K2K Sports …' 'Int. Sport …'
 'Brazil Soccer' 'Toldra Consulting …' 'Hernan Berman' 'Claes Agency'
 'FullNinety' 'Boutique Transfers' 'Zénon Melon' 'firsteleven ISM'
 'Sport Avenir …' 'Extra Time Sports …' 'MARTIN RIHA' 'PabloBueno360'
 '12 MANagement' 'Tim Pole' 'Beckster Int.' 'Simply Sport …'
 'Sports Invest UK ltd' 'Phoenix Sports …' 'Two Touch Agency'
 'Element Twelve' 'Intermedia Sport …' 'Footfeel ISM'
 'G&ML SPORTS LIMITED' 'Fortitude Sports …' 'Carlos Leite' 'Quinas'
 'A10 PRO' 'Gest Eleven' 'SBM' 'JM10 SPORT' 'Fabryka Futbolu'
 'MV SPORTS & MKT' 'Promoesport' 'Classico' 'Rol Sports …' 'Tier One'
 'CLS Mundial' 'HBR Sport' 'Niagara Sports …' 'BS Group - BS Law' 'V4S'
 'Fútbol División' 'TDS SPORTS' 'Athlete Solution Inc'
 'ANTHONY FINNIGAN …' 'Pedro Aldave Ortuzar' 'YMU Group'
 'Prodigy Sports Group' 'Tenet Football' 'Mondial'
 'UKSPORTMANAGEMENT.CO…' 'ProStar' 'Jorge Pires' 'BTFM' 'Teranga Sports'
```

'11WINS' 'Sport First' 'Sport Cover' 'Léandre Chouya'
'M-Soccermanagement' 'ESHA' 'Footwork' 'FFP Agency' 'FUTBOL21 AGENCY'
'MD Management' 'Stonemountain …' 'Majestyk Sports …' 'Every Aspect'
'PEP Consulting' 'Wonder Sports' 'Major League Sports …'
'Frederic Guerra …' 'Eurosportsmanagement…' 'Pinnacle Sports' 'YPFC'
'Keypass AS' 'InterStarDeporte' 'Forza Sports Group' 'Sports&Life'
'Vigo Global Sport …' 'JCRsport' 'Connect Force' 'Avid Sports Group'
'BLACKSKILL' 'Stirr Associates' "YA'ATS SPORTS" 'StormSportsX'
'Paul Latouche' 'fair-sport GmbH' 'Gonçalo Palhinha'
'Friends & Football …' 'SBE' 'CSM' 'Interplayers' 'Volenti' 'ABADSPORT'
'Godfrey k Torto' 'PSM' 'PRO Profil GmbH' 'BMS Sportconsulting …'
'Manuel García …' 'ATG Sports' 'Carmenta' 'BALLWERK Sports GmbH'
'Neubauer 13 GmbH' 'Team Raiola' 'Agent is known - Player under 18'
'Kögl & Partner GmbH' 'SBE Management AG' 'AKA Global GmbH' 'SWSports'
'Robert Schneider' 'HMH Sportmanagement' 'Epic Sports' 'Thomas Meunier'
'Teamgeist …' 'ISM Int. Soccer …' 'Lorimanagement' 'Kick&Run'
'management 360' 'Solbakken / Player …' 'Marjan Sisic'
'TrueSports GmbH' 'CN Sports' 'MR Sport Management …'
'EM Sports Consulting' 'D20 Sports' 'Prolific Sports …'
'teamplayer …' 'Eleven Talent Group' 'Concept Sport Agency'
'AIS SPORT SERVICE' 'KL Sportsbase' 'PESM' 'Creatio Management' 'Wolfs'
'Sports eXcellence …' 'Skylar' 'Roberto Tukada' 'NSG ApS' 'FTC'
'H.S.P. …' 'B360 Sports' 'Dr. Michael Becker' 'Fröhlich …'
'NINE Group' 'SMI SPORTS …' 'Dr. Marco Gutfleisch' 'NexSt11'
'TPWE GmbH' 'Hexagon' 'Kostila' 'Karl M. Herzog …' 'Stephan Engels'
'BMG-SPORT' 'People In Sport' 'YOU FIRST' 'VIDA 11'
'Apertura Sports GmbH' 'Bigpoint' 'F-F SPORT' 'Sport Business …'
'no agent' 'WINStar Management' 'Elite Sports …' 'Soccer Mondial AG'
'Soccertalk GmbH' 'SMG GmbH' 'FSB - …' 'Pando' 'think forward'
'B O K S' 'SPORT INVEST' 'C.R' 'Mondo Victory' 'Rummenigge Sport …'
'Caligiuri Sports' 'Konstantinos Farras' 'Universal Sports …'
'MK Sport Consulting' 'Klan Soccer FZCO' 'Ulisses Santos - …'
'Haspel …' 'W&K Fairplay' 'ROOF Talents' 'W.Egal O.S.F.M.' 'BY AND FOR'
'PURE' 'MD Sports' 'DC International …' 'Timo Hübers' 'LK Management'
'act SPORTS GROUP' 'NGM SPORTS' 'ProfiSports Prskalo' 'CS'
'GOOD ADVICE Sports …' 'ROGON U23' 'Golden Patch …' 'Ariel Krasouski'
'Ivan Bosnjak' 'Briem Soccer …' 'Excellence Sport' 'BoostMySport'
'Lasana KOITA' 'Quantum Sport' 'feel soccer GmbH' 'DLT Sports Group'
'CMA Group' 'Siebert & Backs' 'ROYAL' 'Key United' 'Wojciech Zlobicki'
'NGA Sport' 'Murza Eduard' 'WoS' 'TDL Sports …' 'UNIDOS'
'IS SPORTS AGENCY …' 'Kay Sports …' 'JEB ENTERTAINMENT …' 'GoalSky'
'BY Sport Consulting' 'Group Athletic …' 'RX BROTHERS' 'Soccer11'
'Roberto Calenda' 'GBG Global Business …' 'M.A.R.A.T. Football …'
'Maxime Nana' 'TMP SOCCER srl' 'ESN' 'Pietro Chiodi …' 'GG11'
'Gr Sports' 'Patrick Bastianelli' 'Unnak Sports' 'DW Sports Management'
'A-GROUP' 'not clarified' 'DLF' 'One Team Football' 'Goal Management'
'QUAN SPORTS …' 'G.E.V. …' 'Epic Team' 'BE Sport Management'
'MVP United' 'TEAM LUKAKU' 'Starbridge Srl' 'Davide Torchia' 'B2F'

'Castelnovo' 'DCGLOBALSL' 'SP' 'PDP s.r.l. …' 'Andrei Martínez'
'Guastadisegno' 'INTERNATIONAL …' 'F-MG.COM' 'Cenk Melih Yazıcı'
'Madeira Sports' 'Link' 'Villarreal, Schena …' 'Gelu Rodríguez'
'Gianni Vitali' 'OmniSports' 'Reset Group Srl' 'H2 Agency' 'BSM'
'Dream and Goal' 'Football Mix B.V.' 'GP Soccer and …'
'CT10 management Srl' 'Manuel El Shaarawy' 'Keeper Football …'
'Avvocato Claudio …' 'Matteo Materazzi' 'Italian Managers …'
'DOO Magnum' 'Mateja Kezman' 'OTB Sports' 'Conquest' 'MM-Management'
'Gea World' 'Attacantti Sports …' 'LUCA DE SIMONE'
'Football Now Limited' 'Prattes Group' 'HAGMAYR …' 'PRO10 Sports' 'SRM'
'Simone Seghedoni' 'Equipo TMA' 'Top Eleven …' 'AKADA' 'Prime11'
'Elite FA' 'GlobalGroup11' 'Sport Profile' 'Shpinev Vadim'
'One Team Management' 'Assist' 'MSC GROUP' "Antoniu's Assessoria" 'KTM'
'Football Service' 'Riccardo Calleri' 'The Kulture Group'
'Iconics Management' 'BSSM' 'Foot & Ball' 'LIOS Sports' 'PR10'
'Regenera Elite' '11MANGMT' 'BC Group' 'UruFutbol' 'SPORTS CONNECTION'
'Sergio Berti' 'Fernando Felicevich' 'Football Promotions …' 'Ballex'
'Sportlex Srl' 'Arna Sports …' 'Carlo Alberto …' 'SP GROUP'
'Football Capital' 'Promanager' 'FF' 'Piccioli&Lombardi' 'BestFoot FM'
'Alberto Jimmy …' 'AR Sport Management' 'Green Soccer' 'ES'
'TMG Football Limited' 'Fedele Management …' 'Promanage'
'Creative & …' 'IDUB GLOBAL' 'Luigi Lauro SBM srl' 'KS LawFirm'
'PIVETEAU' 'Team of Future' 'Marcel Kacinari' 'Daniele Ferri'
'Guido Gallovich' 'Match2gether' 'BS & PARTNERS' 'TOP SPORT MANAGEMENT'
'Kemari' 'Carrierefoot' 'Dedra Football …' 'Wesport Football'
'Mariusz Kulesza' 'PRO TRANSFER' 'Platinum One Group' 'INNfootball'
'topgoal sports …' 'MSSports' 'KFM' 'prosm' 'BHM Sports Agency'
'Pedacus' 'CHJ Management' 'Seven Stars' 'Donato Orgnoni'
'OFP OMNIA FOOTBALL …' 'Zmiya s.r.o' 'Estilo 10' 'FutureBall'
'Amin Sakman' 'MVP Group' 'BSP FOOTBALL AGENCY' 'AGB Sport Management'
'SSM Agency' 'VS Sinergia' 'Stars & Friends' 'Romano Eleven SAS'
'LP Manager' 'PW SOCCER …' 'VV Consulting' 'Christophe Henrotay'
'UJ Football Talent' 'FS Management' 'Ginés Carvajal' 'Un1que Football'
'INTEGRAL ADVISING …' 'Puyol, De la Peña …' 'TACTIC GRUP - …'
'LEADERBROCK' 'Ambition Group' 'Miha Mlakar' 'Gesport Espizua SL'
'Zarko Pelicic' 'Blueprivate …' 'Augustin Jimenez' 'TEAM DEPAY'
'Feel Winner' 'Sparka Management' 'LF Sport Management' 'Primotempo'
'World Soccer Star …' 'MagicPlayers' '10Management' 'AIS Football'
'ADM Esporte' 'JV SPORTS SL' 'OLIMPUS ESPORT …' 'PCR Sports'
'Juzzt Football' 'Carmelo Sánchez' 'Acción Sport' 'Media Base Sports'
'RGFOOTBALL' 'Jesus Medina Martin' 'Free Football' 'Libera Sports'
'INVSPORT (Sergio …' 'ACE' 'MESAS SPORT' 'ADM PRO' 'Master Talents'
'CANTERA LATINA / …' 'starbox' 'AGREF' 'Field Management'
'RR-Soccer …' 'TPSportManagement' 'CONTROL ORIENTADO' 'SPORTBACK' 'FGM'
'La Squadra Sports' 'PASCUAL FERNANDO …' 'JONATHAN SÁNCHEZ' 'Viasport'
'Sports and …' 'GLOBAL ASES' 'Identity Sports' 'Roalza' '7DS'
'Carlos Bilicich' 'MG MEDIATOR TRANSFER' 'LGT Football' 'DV7 Management'
'DRL' 'TRIPLE MATCH' 'Think Ball & Sports …' 'FPD Agency' 'PARRI GROUP'

```
 'MRH Football Agency' 'OKON' 'JC SOCCER' 'Universal Sport' 'HC'
 'Fabián Bustos' 'PROTIO SPORT' 'NTS Management' 'Antonio López …'
 '20 Sport' 'FM11' 'EF11 SL' 'P.P.M.' 'Eugenio Botas …' 'Emartsoccer'
 'REVOLUTION Football …' 'Ramazani Mwanuke' 'Soccersport' 'OFSPORTS'
 'Sport Business' 'Axia Sports …' 'Borja Couce' 'M.AG' 'Faro Sports'
 'PRI Sports' 'Rafael Anguita' '380amk' 'WORLD SPORTS …' '4MB'
 'Ballers Management …' 'EGGO Asesores …' 'MEDIATION CONSEIL …'
 'TRDS' 'INEM SPORTS' 'Sarcle Co. Ltd.' 'DVL Management'
 'GOLDBAEK SPORT GmbH' 'Raps Management' 'B.C Football Agency'
 'Football Solution …' 'SVF FOOT' 'MASALA FOOTBALL …'
 'Prodigy Players' 'Kelyxane Sport …' 'J.Gomes Avocat' 'FIA Football'
 'Marcus Marin' 'E CASTAGNINO' 'GRINTA' 'MERCATO CONSEIL' 'LF360'
 'CH Conseil et …' 'LB SPORT & …' 'Mickael Libong' 'Bucket Group …'
 'MSB Sports' 'Mansa Sports …' 'score agencies' 'Invictus Team'
 'Focus Soccer Agency' 'Henri Zambelli' 'Soccermax' 'Sébastien Cagnet'
 'Team Spirit …' 'Dodici' 'USM GROUP' 'Sferico Sports …'
 '11-Football Players' 'MY ELEVEN CONSULTING' 'Nessim Zeggaie' 'Talentz'
 'H2C Sport Consulting' 'D2PSports' 'Kuniy & W. Sports …'
 'Wilders Sport' 'Alexandre Gontran' 'pro soccer …' 'ASR SPORT'
 'MP FOOTBALL' 'MDC Advisors' 'PSC' 'Quadrans' 'ER Sport Agency'
 'Youssoupha Fall' 'BRAIN VERSE' 'N.Agency' 'Cracksman' 'BMG'
 'Jason N. Pappe' 'Daniel Steinfeld' 'SPORTS ACTION' 'Gold Kick SA'
 'Jamal Saad' 'LEFT Sports' 'R4SE' 'RSport Agency' 'PROSPORT Management'
 'AGJ SP MGMT' 'Power Soccer …' 'World Elites' 'FFP' 'Amc sport paris'
 'Victor Aupetit' '2SAgency' 'Futbol Consultants …' 'Qigsports'
 'Pro Foot Consulting' 'FENOMENOW' '4 THE PLAYER …' 'CNS'
 'Oscar Damiani' 'Elenko Sports Ltda.' 'Alke Sports Group …'
 'MVP Sports …' 'Gallea Gestion S.A' 'Louis DENOLLE' 'Love Players'
 'Optimum Football …' 'ISF SPORTS' 'Supernova Management'
 'Global Football …' 'Alliance Soccer …' 'Just 4 player Group'
 'Talents Sports']
Unique values in column 'outfitter': ['Puma' 'unknowen' 'Nike' 'adidas' 'New
Balance' 'Under Armour' 'Umbro'
 'AB1' 'Sells' 'Uhlsport' 'HashtagOne' 'Stanno' 'Mizuno' 'Lotto' 'Joma']
Unique values in column 'nationality': ['Brazil\xa0\xa0Portugal'
'Germany\xa0\xa0Spain' 'England' 'Portugal'
 "Netherlands\xa0\xa0Cote d'Ivoire" 'Spain\xa0\xa0France'
 'Switzerland\xa0\xa0Nigeria' 'Spain' 'England\xa0\xa0Jamaica'
 'Argentina\xa0\xa0Spain' 'Germany' 'Belgium' 'England\xa0\xa0Ireland'
 'Algeria\xa0\xa0France' 'Norway' 'Argentina'
 'Senegal\xa0\xa0Guinea-Bissau' 'United States\xa0\xa0Poland'
 'England\xa0\xa0Italy' "France\xa0\xa0Cote d'Ivoire"
 'France\xa0\xa0DR Congo' 'Senegal\xa0\xa0France'
 'England\xa0\xa0Sierra Leone' 'Brazil\xa0\xa0France' 'France\xa0\xa0Mali'
 'Switzerland\xa0\xa0DR Congo' 'Croatia' 'England\xa0\xa0Guyana'
 'England\xa0\xa0Nigeria' 'Ukraine' 'United States\xa0\xa0Croatia'
 'Morocco\xa0\xa0Netherlands' 'Albania\xa0\xa0England' "Cote d'Ivoire"
 'Gabon\xa0\xa0France' 'United States\xa0\xa0Lithuania'
```

```
'France\xa0\xa0Cameroon' 'Brazil' 'Poland' 'Scotland\xa0\xa0Isle of Man'
'Japan' 'Ghana' 'Italy\xa0\xa0Brazil' 'Egypt' 'Switzerland'
'Brazil\xa0\xa0Italy' 'England\xa0\xa0Ghana' 'Ireland'
'Netherlands\xa0\xa0Suriname' 'England\xa0\xa0The Gambia'
'Cameroon\xa0\xa0Germany' 'Scotland' 'Greece' 'Spain\xa0\xa0Serbia'
'Spain\xa0\xa0Brazil' 'Guinea' 'Portugal\xa0\xa0England' 'Colombia'
'Netherlands\xa0\xa0Togo' 'Uruguay' 'France\xa0\xa0Martinique' 'Sweden'
'England\xa0\xa0Angola' 'Netherlands\xa0\xa0Curacao'
'England\xa0\xa0DR Congo' 'Brazil\xa0\xa0Spain' 'Scotland\xa0\xa0England'
'Denmark' 'Austria' 'Netherlands' 'Iraq\xa0\xa0England'
'England\xa0\xa0St. Kitts & Nevis' 'Sweden\xa0\xa0Cameroon'
'Uruguay\xa0\xa0Spain' 'France\xa0\xa0Guadeloupe' 'France'
'England\xa0\xa0United States' 'England\xa0\xa0Portugal' 'Wales'
'Denmark\xa0\xa0France' "Mali\xa0\xa0Cote d'Ivoire" 'Senegal'
'Korea, South' 'Netherlands\xa0\xa0Nigeria'
'Sweden\xa0\xa0North Macedonia' 'Slovakia' 'England\xa0\xa0Barbados'
'England\xa0\xa0Scotland' 'Northern Ireland\xa0\xa0England'
'Wales\xa0\xa0England' 'England\xa0\xa0Montserrat' 'Paraguay'
'Sweden\xa0\xa0Eritrea' 'France\xa0\xa0Philippines' 'Morocco'
'France\xa0\xa0Central African Republic' 'Germany\xa0\xa0Burundi'
'Italy\xa0\xa0Nigeria' 'Czech Republic' 'Argentina\xa0\xa0Italy'
"Cote d'Ivoire\xa0\xa0France" 'Italy' 'Jamaica\xa0\xa0England' 'Turkey'
'Australia\xa0\xa0Scotland' 'Denmark\xa0\xa0Germany' 'Northern Ireland'
'England\xa0\xa0Egypt' 'England\xa0\xa0St. Lucia'
'Portugal\xa0\xa0Cape Verde' 'Nigeria' 'France\xa0\xa0Senegal'
'Belgium\xa0\xa0DR Congo' 'Zambia' 'Finland' 'Poland\xa0\xa0England'
'Jamaica' 'Burkina Faso' 'England\xa0\xa0Ukraine'
'Portugal\xa0\xa0Guinea-Bissau' 'Ireland\xa0\xa0England'
'Portugal\xa0\xa0Brazil' 'Spain\xa0\xa0Mali' 'Austria\xa0\xa0Serbia'
'Mexico' 'Ireland\xa0\xa0Nigeria' 'Germany\xa0\xa0Cameroon'
'Brazil\xa0\xa0Serbia' 'Nigeria\xa0\xa0England' 'Mali'
'Spain\xa0\xa0England' 'Canada\xa0\xa0England' 'Ecuador\xa0\xa0Spain'
'Ghana\xa0\xa0England' 'Ecuador' 'Sweden\xa0\xa0Tunisia'
'Ecuador\xa0\xa0England' 'Germany\xa0\xa0Turkey'
'Bosnia-Herzegovina\xa0\xa0Canada' 'England\xa0\xa0Poland'
'Belgium\xa0\xa0Senegal' 'Mali\xa0\xa0France' 'France\xa0\xa0Argentina'
'Costa Rica\xa0\xa0Spain' 'Senegal\xa0\xa0Belgium' 'Ghana\xa0\xa0France'
'New Zealand' 'Montserrat\xa0\xa0England' 'Albania\xa0\xa0Greece'
'Grenada\xa0\xa0England' 'Iran\xa0\xa0Sweden' 'DR Congo\xa0\xa0France'
'Cameroon\xa0\xa0France' 'Germany\xa0\xa0Nigeria'
'Netherlands\xa0\xa0Belgium' 'Dominican Republic\xa0\xa0Spain'
'United States' "Italy\xa0\xa0Cote d'Ivoire"
"England\xa0\xa0Cote d'Ivoire" 'England\xa0\xa0Grenada' 'Serbia'
'France\xa0\xa0Comoros' 'Ghana\xa0\xa0Germany'
"Cote d'Ivoire\xa0\xa0England" 'France\xa0\xa0England'
'France\xa0\xa0French Guiana' 'Ireland\xa0\xa0Northern Ireland'
'United States\xa0\xa0England' 'Netherlands\xa0\xa0Mozambique'
'Portugal\xa0\xa0Germany' 'Australia\xa0\xa0England'
```

'Brazil\xa0\xa0Belgium' 'Israel\xa0\xa0Portugal'
'Ireland\xa0\xa0United States' 'Uruguay\xa0\xa0Italy'
'Zimbabwe\xa0\xa0England' 'Denmark\xa0\xa0Nigeria'
'France\xa0\xa0Guinea-Bissau' 'France\xa0\xa0Spain'
'Canada\xa0\xa0Liberia' 'Croatia\xa0\xa0Germany' 'Germany\xa0\xa0England'
'Germany\xa0\xa0Austria' 'Germany\xa0\xa0Kosovo' 'Germany\xa0\xa0France'
"Germany\xa0\xa0Cote d'Ivoire" 'Germany\xa0\xa0Poland'
'Germany\xa0\xa0Greece' 'Portugal\xa0\xa0France' 'Turkey\xa0\xa0Germany'
'France\xa0\xa0Guinea' 'Germany\xa0\xa0Syria'
'United States\xa0\xa0Portugal' 'Hungary' 'Hungary\xa0\xa0Germany'
'Germany\xa0\xa0Senegal' 'Germany\xa0\xa0Ghana' 'Slovenia'
'Denmark\xa0\xa0Tanzania' 'Finland\xa0\xa0Slovakia' 'Russia'
'Netherlands\xa0\xa0Ghana' 'Germany\xa0\xa0Afghanistan'
'France\xa0\xa0Morocco' 'Germany\xa0\xa0Morocco' 'Iran' 'Albania'
'Portugal\xa0\xa0Angola' 'United States\xa0\xa0Germany'
'Switzerland\xa0\xa0Senegal' 'Germany\xa0\xa0Togo' 'Algeria'
'Denmark\xa0\xa0Italy' 'Germany\xa0\xa0Ireland'
'Luxembourg\xa0\xa0Cape Verde' 'France\xa0\xa0Italy'
'Germany\xa0\xa0DR Congo' 'United States\xa0\xa0Dominican Republic'
'Egypt\xa0\xa0Canada' 'Germany\xa0\xa0Montenegro'
'Nigeria\xa0\xa0Germany' 'Bosnia-Herzegovina\xa0\xa0Germany'
'Denmark\xa0\xa0United States' 'Togo\xa0\xa0Germany' 'Israel'
'Suriname\xa0\xa0Netherlands' 'United States\xa0\xa0France'
'Switzerland\xa0\xa0Dominican Republic' 'Italy\xa0\xa0Germany'
'Italy\xa0\xa0Ghana' 'France\xa0\xa0Malta' 'Denmark\xa0\xa0Ghana'
'DR Congo' 'Guinea\xa0\xa0France' 'Australia\xa0\xa0Southern Sudan'
'Germany\xa0\xa0Angola' 'Luxembourg\xa0\xa0Portugal'
'Switzerland\xa0\xa0Portugal' 'Morocco\xa0\xa0Germany'
'Germany\xa0\xa0Albania' 'France\xa0\xa0Réunion' 'Tunisia\xa0\xa0France'
'Germany\xa0\xa0Bosnia-Herzegovina' 'Luxembourg\xa0\xa0Denmark'
'Germany\xa0\xa0Kenya' 'Greece\xa0\xa0Brazil' 'Armenia'
'Germany\xa0\xa0Hungary' 'Montenegro\xa0\xa0Serbia'
'Germany\xa0\xa0Algeria' 'Germany\xa0\xa0Brazil'
'Serbia\xa0\xa0Switzerland' 'Bulgaria\xa0\xa0Germany'
'France\xa0\xa0Poland' 'Czech Republic\xa0\xa0Slovakia'
'Sweden\xa0\xa0Serbia' 'Germany\xa0\xa0Egypt'
'The Gambia\xa0\xa0Switzerland' 'Costa Rica' 'Cameroon'
'Philippines\xa0\xa0Germany' 'Congo' 'Kosovo\xa0\xa0Albania'
'Germany\xa0\xa0Italy' 'North Macedonia' 'Georgia' 'Romania'
'England\xa0\xa0Canada' 'Germany\xa0\xa0Finland'
'United States\xa0\xa0Netherlands' 'Bosnia-Herzegovina'
'Spain\xa0\xa0Morocco' 'France\xa0\xa0Algeria' 'Belgium\xa0\xa0Kenya'
'Sweden\xa0\xa0Bosnia-Herzegovina' 'Albania\xa0\xa0Italy'
'Netherlands\xa0\xa0Aruba' 'Germany\xa0\xa0Netherlands'
'Albania\xa0\xa0Switzerland' 'Serbia\xa0\xa0Belgium'
'Italy\xa0\xa0Canada' 'Serbia\xa0\xa0Slovakia' 'The Gambia'
'Bosnia-Herzegovina\xa0\xa0Sweden' 'Italy\xa0\xa0Australia'
'Italy\xa0\xa0Egypt' 'Lithuania' 'Serbia\xa0\xa0Spain'

'Ghana\xa0\xa0Italy' "Cote d'Ivoire\xa0\xa0Italy"
'Serbia\xa0\xa0Bosnia-Herzegovina' 'Turkey\xa0\xa0Austria'
'Germany\xa0\xa0United States' 'Equatorial Guinea\xa0\xa0Spain'
'Albania\xa0\xa0Sweden' 'Guadeloupe\xa0\xa0France'
'Switzerland\xa0\xa0Chile' 'England\xa0\xa0Guinea-Bissau'
'Paraguay\xa0\xa0Spain' 'Ireland\xa0\xa0Ghana' 'Morocco\xa0\xa0Italy'
'Slovenia\xa0\xa0Austria' 'Serbia\xa0\xa0Germany' 'Togo'
'Chile\xa0\xa0Spain' 'Bulgaria' 'Portugal\xa0\xa0Luxembourg'
'Croatia\xa0\xa0Bosnia-Herzegovina' 'Albania\xa0\xa0Kosovo'
'New Zealand\xa0\xa0Italy' 'Nigeria\xa0\xa0Netherlands'
'Mexico\xa0\xa0Spain' 'Netherlands\xa0\xa0Angola' 'Cyprus' 'Chile'
'Iceland' 'Morocco\xa0\xa0France' 'Italy\xa0\xa0Venezuela' 'Uzbekistan'
'Angola\xa0\xa0France' 'Latvia' 'Sweden\xa0\xa0Burkina Faso'
'Italy\xa0\xa0Netherlands' 'Australia\xa0\xa0Bosnia-Herzegovina'
'Sierra Leone' 'Bosnia-Herzegovina\xa0\xa0Italy' 'Colombia\xa0\xa0Spain'
'Venezuela' 'Nigeria\xa0\xa0Belgium' 'Italy\xa0\xa0Cameroon'
'Germany\xa0\xa0Sierra Leone' 'France\xa0\xa0Congo'
'Spain\xa0\xa0Netherlands' 'France\xa0\xa0Benin'
'Spain\xa0\xa0Dominican Republic' 'Spain\xa0\xa0Guinea-Bissau'
'Mozambique' 'Central African Republic\xa0\xa0France'
'Belgium\xa0\xa0Martinique' 'Belgium\xa0\xa0Spain'
'Denmark\xa0\xa0Sweden' 'Senegal\xa0\xa0The Gambia'
'Spain\xa0\xa0Argentina' 'Spain\xa0\xa0Algeria' 'Spain\xa0\xa0Ghana'
'Ghana\xa0\xa0Spain' 'Switzerland\xa0\xa0Turkey' 'Spain\xa0\xa0Colombia'
'Guinea\xa0\xa0Spain' 'Morocco\xa0\xa0Canada'
'Netherlands\xa0\xa0Tunisia' 'Serbia\xa0\xa0Netherlands'
'Croatia\xa0\xa0Switzerland' 'Mexico\xa0\xa0Portugal'
'Peru\xa0\xa0Netherlands' 'United States\xa0\xa0Spain'
'Switzerland\xa0\xa0Bosnia-Herzegovina' 'Spain\xa0\xa0Portugal'
'Morocco\xa0\xa0Spain' 'Peru\xa0\xa0Spain' 'North Macedonia\xa0\xa0Spain'
'Ghana\xa0\xa0Nigeria' 'Montenegro\xa0\xa0Argentina'
'Albania\xa0\xa0Spain' 'Colombia\xa0\xa0Venezuela' 'Senegal\xa0\xa0Spain'
'Zimbabwe' 'Spain\xa0\xa0Equatorial Guinea' 'Guinea-Bissau\xa0\xa0France'
'Belgium\xa0\xa0Burundi' 'Morocco\xa0\xa0Belgium'
'Venezuela\xa0\xa0Spain' 'Canada\xa0\xa0Jamaica'
'Paraguay\xa0\xa0Argentina' 'DR Congo\xa0\xa0Belgium'
'Honduras\xa0\xa0Spain' 'Spain\xa0\xa0Nigeria' 'France\xa0\xa0Angola'
'Portugal\xa0\xa0Sao Tome and Principe' 'Senegal\xa0\xa0Germany'
'Switzerland\xa0\xa0Cameroon' 'France\xa0\xa0Tunisia'
'France\xa0\xa0Togo' 'France\xa0\xa0Nigeria' 'Belgium\xa0\xa0Ghana'
'Burundi' 'Netherlands\xa0\xa0Dominican Republic'
'France\xa0\xa0Portugal' 'Sweden\xa0\xa0Senegal' 'Kosovo\xa0\xa0Germany'
'Canada\xa0\xa0United States' 'Comoros\xa0\xa0France'
'Cape Verde\xa0\xa0France' 'France\xa0\xa0Croatia'
'Belgium\xa0\xa0Morocco' 'France\xa0\xa0Burkina Faso'
'Slovenia\xa0\xa0Serbia' 'France\xa0\xa0Armenia'
'Cameroon\xa0\xa0Nigeria' 'France\xa0\xa0Haiti'
'Bosnia-Herzegovina\xa0\xa0France' 'South Africa'

```
 'Sweden\xa0\xa0United States' 'Kosovo\xa0\xa0Sweden'
 'Norway\xa0\xa0Rwanda' 'Australia\xa0\xa0France'
 'Bosnia-Herzegovina\xa0\xa0Netherlands' 'French Guiana\xa0\xa0France'
 'Guinea-Bissau\xa0\xa0Portugal' 'Martinique\xa0\xa0France'
 'France\xa0\xa0The Gambia' 'Honduras' 'Benin\xa0\xa0France'
 'Madagascar\xa0\xa0France' "Scotland\xa0\xa0Cote d'Ivoire"
 'Kosovo\xa0\xa0Norway' 'France\xa0\xa0Ghana' 'Congo\xa0\xa0France'
 'Switzerland\xa0\xa0Italy']
```

–Yorum(15)–

-> Burada kategorik özellikleri dönüştürerek, veri setini makine öğrenimi modellerine giriş olarak uygun hale getiriyoruz.

[22]:
```
columns_to_encode = ['league' ,'foot', 'position', 'club', 'contract_expires',␣
 ↪'joined_club', 'player_agent', 'outfitter', 'nationality']

ct = ColumnTransformer(transformers=[('encoder', OneHotEncoder(),␣
 ↪columns_to_encode)], remainder='passthrough')

df_features_encoded = ct.fit_transform(df_features)

df_features_encoded.shape
```

[22]:
```
(2498, 1802)
```

–Yorum(16)–

-> Burada veri setini eğitim ve test için alt kümelerine bölüyoruz ve her birinin boyutunu belirliyoruz.

[23]:
```
x_train, x_test, y_train, y_test = train_test_split(df_features_encoded,␣
 ↪df_target, test_size = 0.3, random_state=22)

y_train = y_train.values.ravel()
y_test = y_test.values.ravel()

print(f'x_train: {x_train.shape}')
print(f'x_test: {x_test.shape}')
print(f'y_train: {y_train.shape}')
print(f'y_test: {y_test.shape}')
```

```
x_train: (1748, 1802)
x_test: (750, 1802)
y_train: (1748,)
y_test: (750,)
```

–Yorum(17)–

-> Grid Search yöntemini kullanarak XGBoost regresyon modelinin en iyi hiperparametrelerini buluyoruz ve bu parametrelerle en iyi modeli eğitiyoruz.

[24]:
```
param_grid = {'nthread':[4],
              'objective':['reg:squarederror'],
              'learning_rate': [0.03, 0.05],
              'max_depth': [4, 7],
              'min_child_weight': [2,3,4],
              'subsample': [0.5, 0.3],
              'colsample_bytree': [0.7],
              'n_estimators': [300]}

xgb = xgboost.XGBRegressor(objective= 'reg:linear')

grid_search = GridSearchCV(estimator=xgb, param_grid=param_grid,␣
 ↪scoring='neg_root_mean_squared_error', cv = 4)

grid_search.fit(x_train, y_train)

best_params = grid_search.best_params_
print(grid_search.best_estimator_)
```

```
XGBRegressor(base_score=None, booster=None, callbacks=None,
             colsample_bylevel=None, colsample_bynode=None,
             colsample_bytree=0.7, early_stopping_rounds=None,
             enable_categorical=False, eval_metric=None, feature_types=None,
             gamma=None, gpu_id=None, grow_policy=None, importance_type=None,
             interaction_constraints=None, learning_rate=0.03, max_bin=None,
             max_cat_threshold=None, max_cat_to_onehot=None,
             max_delta_step=None, max_depth=7, max_leaves=None,
             min_child_weight=3, missing=nan, monotone_constraints=None,
             n_estimators=300, n_jobs=None, nthread=4, num_parallel_tree=None,
             predictor=None, …)
```

–Yorum(18)–

-> Az önce bulduğumuz hiperparametre aramasının sonuçlarını görselleştirmektir. -> Grid Search yöntemiyle farklı hiperparametre kombinasyonlarının performansı ölçülmüş ve bu ölçümlerin sonuçları grid_search.cv_results_ sözlüğünde bulunmaktadır.

[25]:
```
results = grid_search.cv_results_
params = results['params']
mean_scores = results['mean_test_score']

plt.figure(figsize=(10, 6))
plt.plot(range(len(mean_scores)), mean_scores, marker='o')
plt.xlabel('Parameter Combination')
plt.ylabel('Negative Root Mean Squared Error')
plt.title('Grid Search Results')
plt.xticks(range(len(mean_scores)), params, rotation=90)
plt.show()
```

Grid Search Results
Negative Root Mean Squared Error
−12.9
−13.0
−13.1
−13.2
−13.3
Parameter Combination
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 4, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 4, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 4, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 4, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 4, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 4, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 7, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 7, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 7, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 7, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 7, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.03, 'max_depth': 7, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 4, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 4, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 4, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 4, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 4, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 4, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 7, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 7, 'min_child_weight': 2, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 7, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 7, 'min_child_weight': 3, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 7, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.5}
{'colsample_bytree': 0.7, 'learning_rate': 0.05, 'max_depth': 7, 'min_child_weight': 4, 'n_estimators': 300, 'nthread': 4, 'objective': 'reg:squarederror', 'subsample': 0.3}

–Yorum(19)–

-> En iyi hiperparametre değerlerine sahip bir Regressor modeli oluşturarak, bu modele eğitim verilerini uygulayarak modele uygun bir şekilde eğitim yaptırıyoruz.

[26]:
```
best_xgb = xgboost.XGBRegressor( **best_params)

best_xgb.fit(x_train, y_train)
```

[26]:
```
XGBRegressor(base_score=None, booster=None, callbacks=None,
             colsample_bylevel=None, colsample_bynode=None,
             colsample_bytree=0.7, early_stopping_rounds=None,
             enable_categorical=False, eval_metric=None, feature_types=None,
             gamma=None, gpu_id=None, grow_policy=None, importance_type=None,
             interaction_constraints=None, learning_rate=0.03, max_bin=None,
             max_cat_threshold=None, max_cat_to_onehot=None,
             max_delta_step=None, max_depth=7, max_leaves=None,
             min_child_weight=3, missing=nan, monotone_constraints=None,
             n_estimators=300, n_jobs=None, nthread=4, num_parallel_tree=None,
             predictor=None, …)
```

–Yorum(20)–

-> Son olarak bu XGBRegressor modeli üzerinde test verilerini kullanarak tahminler yapıyoruz ve bu tahminlerin performansını değerlendiriyoruz. -> predict metodu kullanılarak x_test verileri üzerinde tahminler yapılır ve bu tahminler pred değişkenine atanır. -> Ardından, tahminlerin gerçek değerlerle karşılaştırılarak modelin performansı ölçülür. -> mean_absolute_error fonksiyonu kullanılarak ortalama mutlak hata hesaplanır. -> Son olarak hata ekrana yazdırılır.

[27]:
```
pred = best_xgb.predict(x_test)

mae = mean_absolute_error(y_test, pred)
mse = mean_squared_error(y_test, pred)
rmse = np.sqrt(mse)

print('mean absolute error: ',mae)
print('mean squared error: ',mse)
print('root mean squared error: ',rmse)
```

```
mean absolute error:  6.926829096972445
mean squared error:  134.74530216874018
root mean squared error:  11.607984414563115
```

[ ]:
```

```