Exploratory Data Analysis

In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
df = pd.read_csv('Country.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	id	name
0	1	Belgium
1	1729	England
2	4769	France
3	7809	Germany
4	10257	Italy

In [4]:
```
df.describe()
```

Out[4]:

	id
count	11.000000
mean	12452.090909
std	8215.308472
min	1.000000
25%	6289.000000
50%	13274.000000
75%	18668.000000
max	24558.000000

In [5]:
```
df.shape
```

Out[5]:
```
(11, 2)
```

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   id      11 non-null     int64 
 1   name    11 non-null     object
dtypes: int64(1), object(1)
memory usage: 304.0+ bytes
```

In [7]:
```
#Checking for missing null values
df.isnull().sum()
```

Out[7]:
```
id      0
name    0
dtype: int64
```

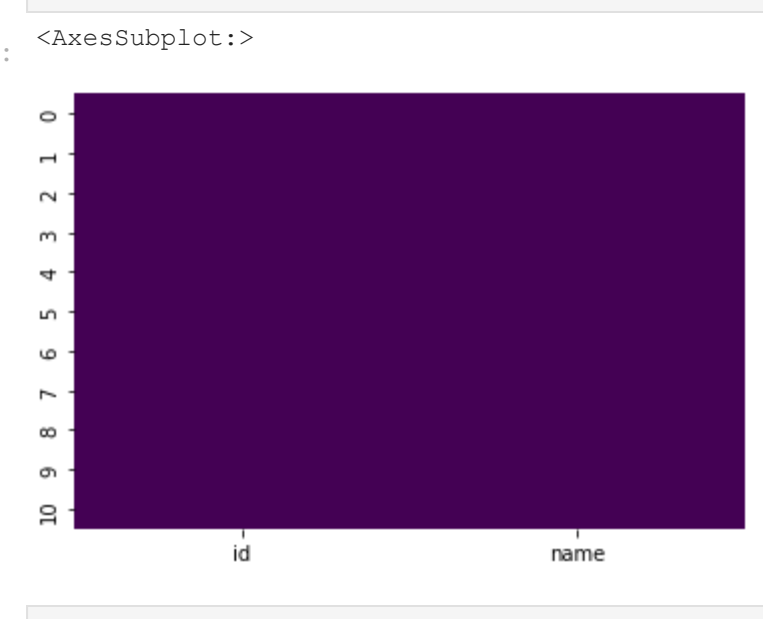
In [8]:
```
#Checking for wrong entries like symbols -,?,#,*,etc.
for col in df.columns:
    print('{} : {}'.format(col,df[col].unique()))
```

```
id : [    1  1729  4769  7809 10257 13274 15722 17642 19694 21518 24558]
name : ['Belgium' 'England' 'France' 'Germany' 'Italy' 'Netherlands' 'Poland'
 'Portugal' 'Scotland' 'Spain' 'Switzerland']
```

In [9]:
```
for col in df.columns:
    df[col].replace({'?':np.nan},inplace=True)
```

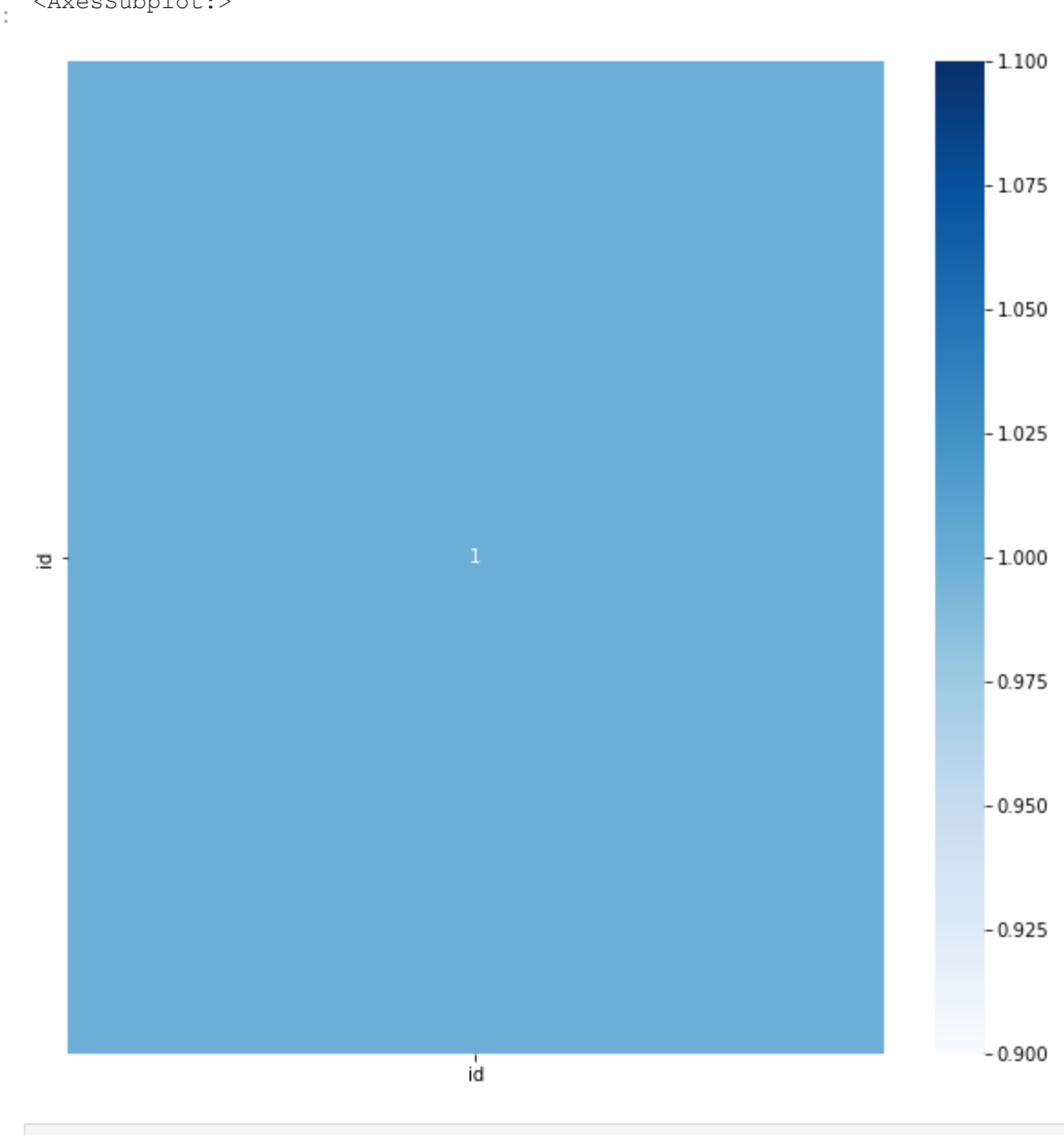
In [10]:
```
sns.heatmap(df.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

Out[10]:
```
<AxesSubplot:>
```



In [11]:
```
plt.figure(figsize=(10,10))
sns.heatmap(df.corr(),cbar=True,annot=True,cmap='Blues')
```

Out[11]:
```
<AxesSubplot:>
```



In [16]:
```
df1 = pd.read_csv('League.csv')
```

In [17]:
```
df1.head()
```

Out[17]:

	id	country_id	name
0	1	1	Belgium Jupiler League
1	1729	1729	England Premier League
2	4769	4769	France Ligue 1
3	7809	7809	Germany 1. Bundesliga
4	10257	10257	Italy Serie A

In [18]:
```
df1.describe()
```

Out[18]:

	id	country_id
count	11.000000	11.000000
mean	12452.090909	12452.090909
std	8215.308472	8215.308472
min	1.000000	1.000000
25%	6289.000000	6289.000000
50%	13274.000000	13274.000000
75%	18668.000000	18668.000000
max	24558.000000	24558.000000

In [19]:
```
df1.shape
```

Out[19]:
```
(11, 3)
```

In [20]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   id          11 non-null     int64 
 1   country_id  11 non-null     int64 
 2   name        11 non-null     object
dtypes: int64(2), object(1)
memory usage: 392.0+ bytes
```

In [21]:
```
#Checking for missing null values
df1.isnull().sum()
```

Out[21]:
```
id            0
country_id    0
name          0
dtype: int64
```

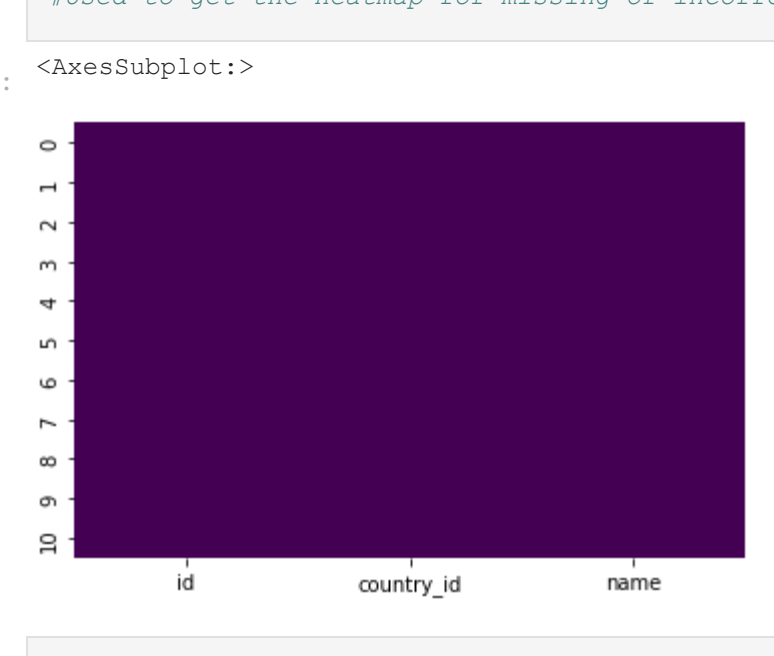
In [22]:
```
#Checking for wrong entries like symbols -,?,#,*,etc.
for col in df1.columns:
    print('{} : {}'.format(col,df1[col].unique()))
```

```
id : [    1  1729  4769  7809 10257 13274 15722 17642 19694 21518 24558]
country_id : [    1  1729  4769  7809 10257 13274 15722 17642 19694 21518 24558]
name : ['Belgium Jupiler League' 'England Premier League' 'France Ligue 1'
 'Germany 1. Bundesliga' 'Italy Serie A' 'Netherlands Eredivisie'
 'Poland Ekstraklasa' 'Portugal Liga ZON Sagres' 'Scotland Premier League'
 'Spain LIGA BBVA' 'Switzerland Super League']
```

In [23]:
```
for col in df1.columns:
    df1[col].replace({'?':np.nan},inplace=True)
```

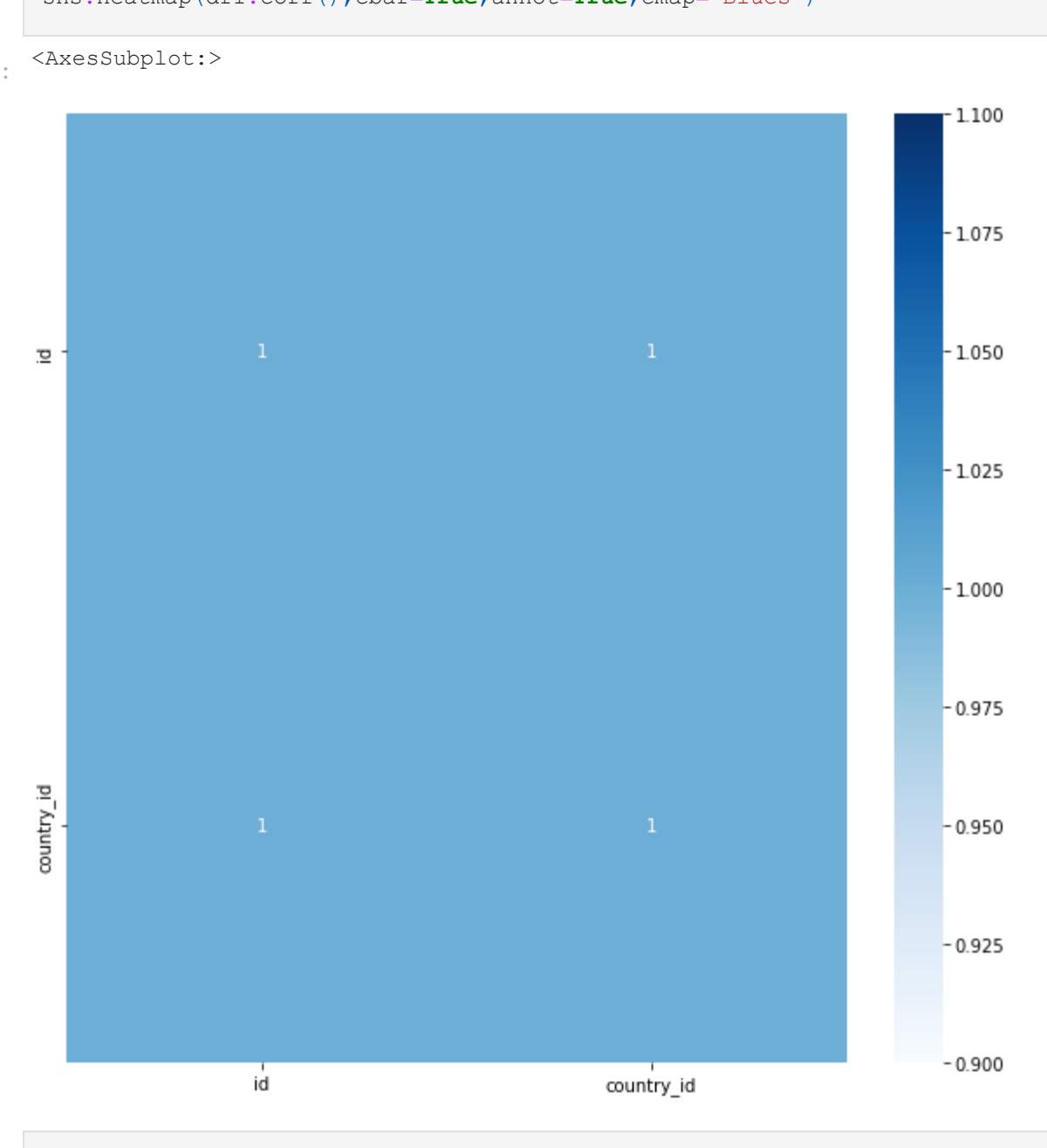
In [24]:
```
sns.heatmap(df1.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

Out[24]:
```
<AxesSubplot:>
```



In [25]:
```
plt.figure(figsize=(10,10))
sns.heatmap(df1.corr(),cbar=True,annot=True,cmap='Blues')
```

Out[25]:
```
<AxesSubplot:>
```



In [15]:
```
df2 = pd.read_csv('Match.csv')
```

In [16]:
```
df2.head()
```

Out[16]:

	id	country_id	league_id	season	stage	date	match_api_id	home_team_api_id	away_team_api_id	home_team_goal	...	SJA	VCH	VC
0	1	1	1	2008/2009	1	17-08-2008 00:00	492473	9987	9993	1	...	4.00	1.65	3.4
1	2	1	1	2008/2009	1	16-08-2008 00:00	492474	10000	9994	0	...	3.80	2.00	3.2
2	3	1	1	2008/2009	1	16-08-2008 00:00	492475	9984	8635	0	...	2.50	2.35	3.2
3	4	1	1	2008/2009	1	17-08-2008 00:00	492476	9991	9998	5	...	7.50	1.45	3.7
4	5	1	1	2008/2009	1	16-08-2008 00:00	492477	7947	9985	1	...	1.73	4.50	3.4

5 rows × 115 columns

In [28]:
```
df2.describe()
```

Out[28]:

	id	country_id	league_id	stage	match_api_id	home_team_api_id	away_team_api_id	home_team_goal	away_tea
count	25979.000000	25979.000000	25979.000000	25979.000000	2.597900e+04	25979.000000	25979.000000	25979.000000	25979.
mean	12990.000000	11738.630317	11738.630317	18.242773	1.195429e+06	9984.371993	9984.475115	1.544594	1.
std	7499.635658	7553.936759	7553.936759	10.407354	4.946279e+05	14087.453758	14087.445135	1.297158	1.
min	1.000000	1.000000	1.000000	1.000000	4.831290e+05	1601.000000	1601.000000	0.000000	0.
25%	6495.500000	4769.000000	4769.000000	9.000000	7.684365e+05	8475.000000	8475.000000	1.000000	0.
50%	12990.000000	10257.000000	10257.000000	18.000000	1.147511e+06	8697.000000	8697.000000	1.000000	1.
75%	19484.500000	17642.000000	17642.000000	27.000000	1.709852e+06	9925.000000	9925.000000	2.000000	2.
max	25979.000000	24558.000000	24558.000000	38.000000	2.216672e+06	274581.000000	274581.000000	10.000000	9.

8 rows × 105 columns

In [29]:
```
df2.shape
```

Out[29]:
```
(25979, 115)
```

In [30]:
```
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25979 entries, 0 to 25978
Columns: 115 entries, id to BSA
dtypes: float64(96), int64(9), object(10)
memory usage: 22.8+ MB
```

In [31]:
```
#Checking for missing null values
df2.isnull().sum()
```

Out[31]:
```
id                0
country_id        0
league_id         0
season            0
stage             0
              ...  
GBD           11817
GBA           11817
BSH           11818
BSD           11818
BSA           11818
Length: 115, dtype: int64
```

In [33]:
```
#Checking for wrong entries like symbols -,?,#,*,etc.
for col in df2.columns:
    print('{} : {}'.format(col,df2[col].unique()))
```

```
 1.87  8.5   1.47  2.29  2.04  9.5   4.6   1.09 13.   17.   15.    1.13
 1.07  1.77  1.26  1.86  3.9   1.06  1.05  2.35  4.25  2.65  2.63  5.2
 6.75 14.   20.    1.21 21.    3.7   1.04 19.   26.   23.  ]
LBD : [ 3.3   3.2   3.6   3.4   3.25  4.5   4.33  4.    3.1   3.5   3.    5.5
 3.75  5.     nan  6.    4.75  3.8   4.2   6.5   2.25  5.75  3.9   3.7
 5.25  7.    5.2   4.6   4.1   2.88  8.   10.    7.5   9.    4.8   2.9
 2.8   2.87  2.3   2.4   2.75  2.5   8.5   6.25  4.3  11.   12.    2.6
 1.8   1.9   2.1   1.91  1.65  2.7   2.2   2.38  1.73  2.    1.4   1.67
 1.44  2.62  1.83  4.4   4.25  9.5  13.    2.85  4.59 10.5  15.   17.
19.  ]
LBA : [ 3.75  3.5   2.5   6.5   1.72  1.62  3.6   2.75 10.    9.    2.38  4.5
 2.8   1.5   1.73  7.    2.2   4.    8.    3.4   1.8   3.2   2.1   3.
 2.6   6.    5.    2.7   1.61  1.66  2.62 12.    2.25  5.5   1.44  4.33
 2.37  3.25  2.   11.    1.67  2.4   2.88  1.57   nan  1.83  2.3   1.53
 1.9   2.87  1.91  1.2   1.4   1.85  8.5   4.2  13.   17.    1.29  3.8
 7.5   1.6   3.1   9.5   1.95  2.9  14.    4.1   1.75  3.3   3.7  15.
 4.75  2.15  2.05  5.75  5.25  1.48  1.36  1.33  1.3  21.   19.   26.
 1.25 23.    1.7   4.8   1.22 10.5   2.29  1.45  4.6   2.55  1.55  1.65
 3.9  16.    2.45  2.63  4.59 29.    1.35 34.    2.04  1.28  1.16  1.17
41.   51.    1.18  1.1   1.77  1.76  4.25  2.35  2.65  5.2   2.85 18.
20.   31.   36.    4.3   1.38  1.14 11.5   1.12  1.13]
PSH : [  nan  5.1    2.48   1.83   1.74   1.58   1.29   1.37   1.76   3.07
  3.15   2.19   1.79   1.65   2.09   2.63   1.24   1.53   1.55   1.99
  4.52   1.89   3.02   2.35   1.59   1.5    1.86   7.74   2.53   1.78
  3.13   1.6    1.92   1.7    2.36   1.54   5.04   2.16   2.25   1.71
  1.56   2.14   3.24   5.42   1.94   1.64   2.67   2.7    2.32   2.15
  3.28   1.33   2.93   2.44   2.18   3.61   4.85   3.34   1.43   1.96
  6.02   1.95   1.62   1.3    2.97   1.57   3.17   2.54   4.63   1.36
  1.38   1.46   4.36   2.55   2.52   5.13   6.     2.78   3.9    5.22
  3.42   1.48   2.3    3.35   2.65   3.49   1.41   2.39   1.77   1.52
  1.28   2.72   2.58   4.2    1.45   1.4    2.71   2.77   2.03   5.35
  2.46   4.57   2.69   2.4    1.67   2.22   1.47   3.18   1.81   1.44
  3.71   2.29   2.31   2.68   6.94   1.22   1.66   2.1    4.07   2.98
  1.93   2.13   4.31   2.     1.34   4.12   2.56   2.27   2.62   1.85
  2.95   1.68   3.97   4.06   2.75   1.39   4.35   2.02   3.76   5.01
  5.37   1.87   1.88   4.23   2.01   1.97   2.79   1.42   3.29   2.88
  1.61   2.45   3.82   3.92   2.61   1.63   1.98   3.04   2.59   4.45
  3.95   3.12   2.64   1.23   2.51   3.41   3.     1.84   2.84   3.69
  3.4    4.67   4.49   2.07   4.28   3.21   3.26   2.38   2.33   1.19
  1.27   3.77   4.43   2.8    4.26   2.66   4.04   2.92   4.91   2.11
  2.74   2.06   2.83   1.72   5.19   2.37   6.71   2.08   1.69   8.37
  1.21   2.04   4.64   2.57   8.66   1.2    5.38   3.19   1.49   3.53
  7.41   3.3    2.91   2.34   3.32   4.29   1.51   1.32   3.67   5.68
  4.1    3.66   2.41   2.23   1.9    7.81   2.89   4.81   4.3    2.86
  4.11   2.43   2.73   6.96   3.05   4.47   3.75   4.01   4.9    2.17
  4.34   5.67   2.42   1.8    2.99   3.88   5.09   2.9    4.89   5.45
  3.46   2.76   1.31   4.46   3.47   2.21   1.73   2.28   3.55   2.12
  3.79   3.2    4.71   2.82   5.66   3.51   4.54   2.96   1.25   3.44
  3.09   6.6    2.26   7.84   1.35   4.18   2.5    3.94   6.45   2.05
  4.62   3.59   4.22   3.03   3.39   3.85   6.26   6.83   1.26   1.91
  3.7    5.96   4.27   1.82   4.39   6.05   5.18   7.1    7.25   4.53
  3.89   5.06   1.18   3.57   7.75   4.21   2.24   5.05   3.23   5.36
  5.5    1.15   5.34   3.14   4.78   3.62   7.26   2.2    2.49   6.52
  3.31   5.24   6.13   4.4    1.75   3.93   4.65   7.48   7.52   5.73
  4.     3.56   5.65   4.59   3.78   6.66   3.65   7.62   4.48   4.87
  5.71   6.25   3.06   3.8    1.16   5.03   3.38   4.19   7.35   8.35
  4.68   5.9    4.6    6.4    4.5    5.3    3.52   3.01   3.98   6.2
  5.16   6.79   5.93   3.36   6.32   8.97   3.48   7.05   6.76   6.09
  6.42   2.85   5.31   6.59   2.47   3.33   9.03   6.58   2.94   1.13
  4.03   6.51   7.99   5.63   6.43   5.98   3.16   3.63   3.87   6.95
  5.89   9.51   8.     1.17   3.22   6.89   7.44   3.58   5.97   5.
  7.65   6.62   3.11   9.27   4.41   9.63   9.37   4.72   4.86   9.19
  8.91   4.95   4.25   8.38   2.6    3.27   5.79   3.45   6.87   5.62
  6.46   5.85   4.8    4.55   5.7    9.85   9.77   5.8    3.96   7.14
  5.54   3.72   3.08  10.2    2.87   7.23   6.88   5.55   6.77   6.21
  6.65   4.83   3.25   4.58   4.88   6.29   7.17  10.8    5.64   4.93
  5.58   4.14   4.24   6.11   8.52   6.15   3.5    5.6    9.2    4.15
  5.52   4.66   4.56   8.05   8.5    5.21   5.75   3.91   4.02   8.15
  4.79   6.68   5.25   4.75   6.24   3.37   3.6    5.51   5.14   2.81
  7.5    4.32   3.84   4.44   7.27   3.64   6.61   5.39   7.91   4.37
  4.51   6.55   4.08   4.7    8.34   7.42   4.05   3.1    5.46   5.82
  9.     5.02   3.73   8.36  10.27   5.33   9.68   3.81  16.92   9.82
  4.96   5.27  11.02   7.33   3.99   9.48   9.1    4.74   7.3    7.87
  6.18   4.97  10.47   5.84  10.88  11.07   7.98   7.94   6.01   4.38
  8.31   3.43   9.14   4.84   4.42   4.13  10.3    3.74   6.36   5.07
 11.25   4.98   7.32   3.68   1.14   7.28   3.83   7.29   5.41   6.64
 10.76   7.58   1.1   11.96   7.12   7.95   1.09   8.55   5.53  13.
  5.77   4.73   7.09   4.61   8.95  10.71   8.69   9.29   8.99  13.95
  7.15   8.4    6.17   8.61  14.24   7.83   4.09   6.37   5.2    7.68
  6.1   11.     8.25   1.11  11.9    1.12   1.07   5.88  13.51   4.17
 12.45   7.69  11.95   3.54  12.8    8.74  14.35  13.26   8.63   4.16
 10.78   9.34  15.25   8.84  13.6    9.7    5.81   5.15  14.15  18.1
  9.79  16.6    5.95   5.49   7.07   5.11  15.5   22.3   15.17   7.03
 10.55   4.82   5.56   5.4   11.8    9.92   8.82  19.24  12.9    9.55
  6.82  17.1   16.2    4.77  10.19  13.83   1.08   8.3    8.39  10.85
  6.84   8.71   7.06  16.77  18.7   13.31  15.     6.74  12.22   5.92
  9.74   6.56   6.7    6.27   9.78   1.909  5.94  23.6   18.5   20.
  4.33   9.76   4.69   8.64  10.33   8.42   8.68   9.46   7.04   6.5
  3.86   8.7    7.88   4.99   9.3    8.01   9.36  10.26   9.15   5.76
 16.43   5.28   5.99  11.93  10.37   5.12   7.46   8.75   4.76   8.03
  6.93   7.7   13.91   8.07   6.54   7.71  14.85   5.48   7.64  12.72
  6.99   6.19   6.41   5.17   6.57   7.82   7.16   5.59  12.5    9.5
  9.44   6.22   7.67  11.28   6.34   7.47   9.45  11.26   6.12   7.01
  6.8    5.87   5.83   6.69   8.24   9.72   6.06   6.78   5.29   6.08
  4.94   8.48  11.15  10.25   8.2    5.26  11.01   5.57   8.67   6.48
 10.02   6.75   5.23   5.69   7.86   6.33  10.79   8.23  20.5    6.9
 10.56  13.58  15.39   5.74  11.3   11.7   10.53   9.75  10.    13.5
 10.82  12.    13.59   9.66   7.19   8.85  13.65   9.54   6.3   14.4
  9.24   9.52  14.     8.93   7.8    6.14   5.08  12.1    5.91  12.48
  7.89   8.89  15.77   8.8   11.69   5.43   7.93  13.43  14.49  12.74
 26.     7.02   9.38   8.78  15.9   17.51   8.47   9.96  15.66   9.91
  8.14  10.61  15.3   16.8   19.    14.79  11.11   6.49   8.29   9.64
 11.29  12.47  25.5    6.81   7.96   7.72  11.6    8.08  13.81  10.65
 10.08  10.7   11.55   9.07  15.1   11.21   9.53  10.93   9.98  14.3
 11.62   9.62  17.52  11.82  14.9    7.66   6.63   9.11   7.08   8.32
 11.5    7.34   8.92   7.13   9.59  10.09   1.188  1.794  1.862  7.76
  9.43  11.33   6.73  10.39  12.36  14.76  12.2    6.44   9.28   6.92
 11.77  10.69  12.77   9.08  10.96  13.48  13.4    7.59  10.24  12.18
 11.45  11.24  13.7   15.37  10.81  11.04   9.41   7.78   8.1   11.19
 10.92  11.1   11.39  11.22  12.65  11.42  13.78   8.59   5.47   8.81
 19.86   9.25  13.07   8.22   1.06  16.19  12.62   1.05   9.61   8.56
 12.43  14.06   7.2   16.81  14.2   16.85  18.09  10.03  14.45  21.08
  6.04  17.05   6.07  16.4   21.2   10.21  17.    14.17  23.    16.
 20.85  22.4   12.24  36.    11.58  18.43   5.44  10.9   18.07  12.98
 11.4    6.97  10.05  21.1    6.85  15.2   11.08  18.95   7.45  18.
 16.56  13.2   11.2   15.6   25.51   1.04  18.89  13.27  31.13 ]
PSD : [  nan  3.82  3.52  3.79  4.27  5.74  5.16  3.93  4.    3.64  3.63  3.81
  4.01  3.51  3.56  6.71  4.6   4.44  3.76  3.88  3.59  3.57  4.4   4.79
  4.39  4.04  4.7   3.95  3.58  4.37  4.02  3.7   4.43  3.87  3.62  3.96
  4.46  3.71  4.08  3.68  3.97  4.26  4.38  4.3   3.47  3.61  5.65  3.49
  3.67  4.2   3.66  4.11  3.69  5.35  4.9   4.58  3.73  4.18  4.53  6.
  3.48  4.42  3.55  3.54  5.32  5.2   4.72  3.35  4.14  3.89  5.94  4.31
  3.84  3.65  4.65  3.3   4.95  3.5   3.53  6.03  4.35  4.85  5.09  3.45
  3.75  4.05  3.6   3.74  3.94  5.08  3.36  4.52  7.19  3.44  3.72  4.28
  5.41  4.15  3.34  5.45  3.4   5.58  3.9   6.2   4.03  4.24  3.85  5.13
  4.99  4.12  3.8   3.41  4.84  3.46  3.29  4.98  6.67  4.33  3.43  4.25
  4.22  3.86  4.09  4.54  4.1   3.31  6.23  4.47  5.28  7.63  6.09  3.77
  4.73  3.78  4.62  5.95  3.92  3.32  4.86  7.15  3.19  4.91  7.64  3.39
  4.06  4.82  7.75  4.67  3.38  5.43  5.25  4.5   5.52  3.42  4.74  7.42
  5.66  5.68  6.02  5.55  4.69  7.28  4.51  4.19  3.37  4.88  7.52  3.83
  6.82  4.57  5.04  4.63  3.91  5.44  6.75  4.29  4.45  4.93  6.05  6.87
  4.64  4.55  4.36  4.17  7.25  6.55  3.21  3.2   6.69  5.03  3.22  5.72
  4.68  7.41  4.34  6.06  4.07  3.98  4.48  6.42  5.76  4.76  4.16  7.5
  4.56  7.49  3.99  8.2   6.68  4.8   4.83  3.26  6.76  9.64  3.25  4.59
  6.8   3.24  5.6   4.94  5.83  7.2   5.05  5.77  5.4   5.75  6.66  7.02
  4.78  5.47  5.26  5.73  6.07  6.15  7.65  5.15  9.07  5.42  5.5   3.33
  6.54  6.38  4.61  6.16  4.96  5.62  5.9   3.12  6.77  5.69  6.22  5.01
  7.11  5.99  4.77  5.48  5.79  5.18  7.4   5.86  7.9   6.25  5.06  4.32
  4.21  3.27  4.13  7.    6.26  3.28  7.7   3.18  7.8   5.    5.21  4.49
 10.4   4.75  6.32  6.9  10.2   3.17  7.45  6.88  9.12  5.3   6.01  8.6
  5.29  8.65 11.03  5.89  8.04  5.23  3.16  3.23  9.47  5.92  5.8   4.71
  7.73  8.1   5.51  6.45  6.29  7.12  5.61  5.02  8.03  8.8   5.93  7.04
  5.87  4.23  4.81  5.1   6.21  6.64  5.36  5.12  4.41  7.92  5.7   6.4
  5.56  8.55  4.92  8.    6.14  5.98  3.14  6.63  5.85  5.46  5.33  6.33
  7.05  4.87  5.07  6.47  5.34  3.04  3.13  7.08  6.35  5.91  6.28  9.35
  6.12  3.08  3.11  3.15  3.1   4.66  8.5   7.61  3.09  7.69  2.93  2.96
  3.07  3.    4.97  8.75  3.06  6.61  6.89  5.63  3.02  5.19  9.2   6.3
  6.34  6.37  5.31  2.94  5.96  5.39  2.97  7.51  4.89  2.98  2.95  3.03
  6.46  9.    2.99  5.38  3.01  5.24  6.65  6.11  8.7   9.98  7.13  9.96
  6.08  3.05  5.64  2.9  11.5  10.   12.    7.83  5.22  7.09  5.11  6.81
  5.88  8.46  8.11  6.36  8.26 12.3   7.26  6.31  6.6   5.71  9.68  6.85
  7.31  5.78  7.79  5.14  8.91  5.57  6.59 10.08 10.87 15.76 13.85 11.07
  9.42  6.74 12.5   6.44  7.93  6.98  8.61 12.17  8.47  7.14  6.1   6.56
 10.1   8.45 12.14  9.04  6.78 13.5   7.38  9.8   6.92 12.55  5.59  8.4
  6.93  5.37  5.81  7.85  6.51  8.95  6.5   7.27 10.33 11.7   7.21 12.87
  8.05  8.52 12.4   5.97  5.27 13.    7.68  7.29  7.3   8.3  14.9   9.9
 10.3  12.6   7.06  5.53  6.86 14.3   9.75  6.13  9.5   6.79  5.67  7.01
  7.84  2.89  2.5   6.04  8.35  2.62  2.2   7.62  7.48  7.6   5.82  7.66
  6.91  9.21  6.24  7.33  5.17  8.25  5.84  8.82  7.32  9.67  6.7   6.52
  8.57  8.77 12.19  8.18  6.27  6.73  6.83  5.54  9.37  8.71  8.07  6.84
  8.41  6.19  8.66  7.78  5.49  6.18  7.94  7.46  8.02  6.62  6.39  7.07
  8.92  8.54  6.17  7.44  8.74  6.95 10.9  10.76 14.06  7.54  6.53  8.28
  8.86  7.1   6.43  7.47  8.06  7.99  7.81 12.83  7.35  8.69 11.79  8.15
  7.39  7.55 15.    8.76  8.81  7.97  8.32 12.2   9.09  9.27 10.8   8.67
 11.87 10.7  11.37 10.85 11.96  7.71  6.48  9.85  8.72  7.76  7.24  9.7
  7.36  9.53 10.01  6.58  8.31 13.19 13.3   6.57  8.08 12.1   9.08 13.34
  7.95 10.47 10.52  9.06 16.75 10.51 14.    9.45  6.72 14.65 18.88 15.29
 20.36 18.34 18.37 22.68 18.31 19.   15.99  8.48 11.39  8.23 13.15 16.85
 10.39 17.49  7.34 13.52 14.32 19.8  13.4  22.5  11.6  19.75 16.    8.9
 17.4  25.   13.25 19.08 10.84 13.49  8.19 15.34 15.5  14.99  9.15 11.1
 18.8  11.45 20.5  11.4  15.75 12.22  8.68 20.    8.94  9.25 10.49 29.
 13.64 16.5  12.8   9.55  9.84 11.  ]
PSA : [  nan  1.76  2.96 ... 13.21  1.13 31.5 ]
WHH : [ 1.7   1.83  2.5   1.44  4.2   2.7   2.2   1.35  1.25  2.6   1.8   2.75
  5.25  4.    1.3   2.3   3.    1.4   1.9   2.05  3.25  2.15  3.2   2.1
  2.25  1.45  1.5   1.6   1.33  1.36  5.    1.22  1.65  1.57  6.5   2.4
  2.    1.73  1.67  1.17  2.9   2.8   1.62  3.8   1.12  4.33  4.5   1.91
  2.38  2.62  4.8    nan  5.5   1.29  3.75  3.6   1.53  1.95  1.2   3.5
  9.    7.    2.65  1.55  6.    1.14  1.85  2.87  1.75  3.3   3.1   1.61
  7.5   1.18  2.88  3.4   2.45  2.55  2.35  1.11  8.    2.37  4.6   1.15
  4.4   4.75 11.    8.5   9.5  10.5   3.9  10.    1.1  12.    1.28  3.7
  1.72  4.1  15.    1.08  4.35  4.25 13.    6.2  19.   17.    5.2   3.15
  1.07 21.    1.16  1.13  1.06  1.05  1.04  1.66  4.9   2.85  3.05  7.25
  3.45  5.1   1.02  5.7  23.   26.  ]
WHD : [ 3.3   3.25  3.75  3.4   4.2   4.5   3.1   3.2   3.5   4.    4.8   4.3
  3.8   3.6   3.15  5.    6.    6.5   5.5    nan  4.33  5.3   3.9   5.1
  3.    4.6   7.5   4.75  2.9   5.25  5.75  7.    8.    4.1   9.    8.5
  9.5   4.4   2.8   2.88  2.87  2.95  2.85  2.75  2.5   3.7   2.7   1.02
  5.8  10.    2.6   1.8   1.83  2.05  1.6   1.95  1.91  1.67  2.62  4.25
  2.1   2.25  2.2   2.4   1.7   1.4   1.73  1.53  1.2  11.    3.45  3.05
  3.35 12.    6.2  13.   15.    2.38 17.  ]
WHA : [ 4.33  3.6   2.4   6.    1.7   2.25  2.75  7.    9.5   3.75  4.2   2.8
  1.55  1.75  8.5   2.6   2.1   3.25  3.    1.95  9.    3.5   2.    2.7
  6.25  7.75  5.75  5.    3.2   2.62  2.9   7.5   8.    1.57  1.6   5.5
  4.5   1.44  2.5   4.8   2.3  10.    2.2   3.1   2.38  1.8  12.    1.67
  1.62  3.4   3.8    nan  1.53  1.83  1.73  3.3   4.    1.65  2.15 11.
  1.9   1.2   1.61  1.91  1.4   2.65  1.45 10.5   6.5  15.    2.37 13.
  2.05  2.88  1.5   1.36  4.6   1.85  2.45 17.    2.55  2.87  4.75  4.4
 19.    1.3   1.33  5.25  3.9  21.    1.25  1.28 23.    1.22 26.    3.15
  3.05  3.7   4.1   2.85  2.35  5.2  34.    1.29  3.55  2.95  1.35  5.4
 29.    1.15  6.8   5.7   6.2   1.12  4.3   5.8   1.14  1.17  1.18  1.72
  3.65  3.45  4.7   4.35  5.3   3.35  4.25 14.    5.1   4.05  6.9   6.6
  5.6  51.   41.    1.1   1.08]
SJH : [ 1.9    1.95   2.63   1.44   4.5    5.5    1.91   2.6    2.2    1.27
  1.8    2.8    1.7    2.3    5.25   4.25   1.33   2.4    3.3    1.85
  1.4    1.83   2.05   3.6    2.15   2.1    3.5    2.25   4.33   1.36
  4.75   1.62   2.38   1.67   6.     1.2    3.2    3.4    1.5    6.5
  1.57   1.73   3.     2.5    1.18   3.25   2.75   1.25   5.     2.35
  1.22   2.     1.75   3.05   6.25   1.29    nan  4.2    4.     4.8
  1.17   3.1    1.14   1.53   3.75  11.     1.45   1.6    7.     4.65
  1.3    2.88   8.     3.8    1.65   1.55   1.15   2.9    2.55   2.7
  1.42   8.5    1.88   3.13   2.45   6.75   3.88   1.48   5.2    3.9
  9.     7.5    3.12   1.571  1.909  1.727  2.375  1.533  2.62   5.8
  4.1    9.5   15.    12.     1.13  10.     1.11   1.87  10.5    5.75
  6.15   1.38   1.23   2.65   1.47   1.64  13.     1.35   1.08   1.86
  1.41  17.     1.1    1.06   1.09   4.6    3.15  21.    14.     1.04
  1.05   1.07   6.2    8.25   3.7    1.12  23.    19.   ]
SJD : [ 3.3    4.     3.5    3.75   3.4    5.     3.25   3.2    4.5    4.33
  5.5    3.6    6.5    3.8    5.75   5.25   6.     3.15    nan  3.1
  3.35   4.2    3.65   3.45   7.5    3.7    4.75   4.8    6.75   7.
  4.6    3.9    3.     5.8    6.25   3.13   4.1    5.2    8.     9.
  8.5    3.33   2.88   2.8    2.75   2.3    2.4    2.9    3.05   3.12
  2.5    4.25  10.     1.73   5.6    1.5    2.1    2.25   1.91   2.2
  2.63   2.     1.4    1.83   1.62   2.7    1.8    2.375  4.4    3.85
 13.     9.5   11.     2.6    6.2    5.4   12.     5.1   14.    15.   ]
SJA : [ 4.     3.8    2.5    7.5    1.73   1.67   3.6    2.4    3.1   10.
  4.25   2.38   4.75   2.88   1.62   1.8    9.     2.75   2.1    7.
  3.5    2.6    1.95   3.3    3.4    3.2    1.83   2.8    8.5    5.5
  3.     8.     5.     3.25  13.     5.25   2.25   2.15   6.5    1.5
  6.     2.3    3.75  12.     2.05   2.2    4.5   11.     1.7    2.
  2.95   4.1    1.53    nan  4.33   2.7    9.5    1.75   2.63   1.57
 14.    16.     1.18   1.45   1.44   1.85   1.9   15.     5.75   4.8
  4.6    2.45   2.33   4.2    1.91   2.55   2.91   3.7    1.4    8.75
  5.2    2.9   17.     1.65   3.9    1.33   2.62   5.8    2.875  1.727
  3.13   3.12   6.25  19.     1.3    1.36   5.3    1.25  21.     1.29
 29.     1.87   1.6    1.47   3.33   6.75   2.65   3.45   1.2   23.
  1.86   1.88  26.     1.27  34.     1.22   1.1    1.55   1.64   1.571
  1.167  1.615 18.     1.17   1.13  12.5    5.6   41.     1.38  25.
  1.56   1.15   8.25 ]
VCH : [ 1.65   2.     2.35   1.45   4.5    4.35   2.1    2.8    2.25   1.3
  1.28   2.65   2.75   1.7    2.2    5.5    3.6    3.25   1.75   1.35
  1.8    2.4    3.75   1.25   3.2    4.     2.5    1.5    1.6    1.4
  5.     1.55   1.2    2.3    2.6    3.     1.95   6.5    2.7    2.9
  1.33   1.85   1.15   1.9    1.83   2.05    nan  3.4    1.91   1.22
  6.     1.18  10.     1.14   1.57   1.44   3.1    1.53   2.62   2.55
  2.38   1.73   1.62   3.5    1.29   4.33   1.67   2.88   4.75   1.36
  7.5    1.17   2.15   3.3    4.2    3.8    3.12   4.6    5.25   7.
  1.12   5.4    1.364  1.444  1.667  1.615  4.333  9.5    4.3    4.4
  1.87   5.75   2.45   4.8    4.1    3.9    3.7    1.92   6.25   8.
  1.47   5.2    3.13   2.63   1.93   3.35   3.15   9.     1.72  11.
  5.3    2.85   6.2    6.75   7.2    8.5   15.     1.11   1.09   1.1
  2.875  1.571  1.909  1.13  10.5   13.     1.533  1.286  2.375  1.98
 17.    12.     1.78   1.06   1.07  19.     1.833  1.952  2.625  1.083
 11.5   18.    21.     1.08  20.     4.7    1.05   1.143 26.    23.
  1.333  1.16   1.04   3.125  1.222 29.     1.925 22.     1.03  36.
 31.   ]
VCD : [ 3.4    3.25   3.75   4.35   4.5    3.3    3.2    3.5    4.     3.1
  5.     3.6    5.5    3.8    4.2    3.7    6.      nan  6.5    4.75
  3.     4.33   3.12   4.4    4.6    5.25   7.5    7.     5.4    8.
  8.5    5.75   3.9    5.2    4.3    4.1    4.8    6.25   6.1    3.35
  3.85   3.45   6.7    5.8    4.7    4.9   10.     9.     9.5    2.5
  3.13   2.9    2.8    2.88   2.75   2.25   2.38   2.4   11.5   11.
 10.5   12.    13.     2.65   3.05   1.62   2.     1.73   2.62   1.7
  3.125  2.2    2.3    2.45   2.15  15.    17.    19.     4.333 22.
 18.    23.    26.   ]
VCA : [ 4.5    3.25   2.65   6.5    1.65   1.7    3.     2.25   2.8    8.5
  2.3    3.4    4.35   2.9    1.55   1.9    2.     7.5    4.     2.6
  1.85   9.     3.1    2.1    1.75   6.     2.75   7.     8.     1.6
  5.    11.     2.35   2.7    1.4    2.5    3.2   12.     4.75   2.2
  5.5    2.4    3.75  10.     3.5    3.6    9.5     nan  3.3    1.45
  1.5    1.2    1.8   13.     1.44   2.15   1.57   2.62   4.33   1.67
  1.62   1.91   2.88   2.38   1.95   2.05   1.73   1.83   4.6    3.8
  3.12   5.25   4.2   15.    19.     1.53   4.4   17.     5.4   26.
  1.833  6.25  23.     1.36  10.5    3.9    4.1    4.8    3.7    5.75
  2.45   2.55   4.3   11.5    1.98   5.2    1.87   3.13   1.92   2.63
  1.93   5.8    1.3    4.7    1.35   1.72   1.29   3.55   2.95   3.35
 16.    21.     5.9    3.45   4.9    1.33   1.25   1.22  29.     1.615
  4.333  2.875 20.    18.    22.     3.125  1.364  1.952 31.    36.
 41.     1.18  34.    51.     1.17   1.15   6.75   1.09   1.909  1.444
 14.     1.12   1.14   2.625  1.667  2.375  1.78  67.     1.533 46.
  1.13   1.08 ]
GBH : [ 1.78  1.85  2.5   1.5   4.5   2.8   2.2   1.25  2.6   2.75  1.75  2.3
  5.25  4.1   1.3   3.1   1.8   1.4   1.95  3.6   2.1   1.27  2.    3.2
  4.25  2.45  1.45  1.37  4.6   1.53  1.62  2.25  1.33  1.7   2.15  4.75
  1.6   1.2   1.67  2.35  2.85  2.05  7.    1.55  1.65  3.    2.4   1.35
  2.9   1.18  3.4   2.65  1.57  2.7   2.55  1.73  1.83  6.25   nan  4.
  4.4   1.36  1.15  3.75 10.    1.47  1.9   6.5   1.22  5.5   5.    3.7
  6.    3.3   3.5   4.33  1.72  1.42  1.38  4.2   3.25  1.44  1.48  6.75
  3.15  1.28  1.14  1.87  8.5   2.95  1.91  1.34  5.75  1.26  1.88  3.9
  8.    7.25  1.16  9.    7.5   2.63  3.45  1.77  1.17 11.    1.13  3.8
  1.1   1.12  9.5   1.23 12.    7.75  2.38  3.65  3.55  1.32  3.05  1.19
  1.63  3.35  4.7   4.3  13.    9.25  1.11 10.5   2.07  1.31 17.    3.85
  1.58  1.43  5.4   1.93  8.75  5.3  21.    4.9   5.05 16.    1.82  1.52
  1.05  1.08  1.09  9.75  1.06  1.56  1.89 11.5   8.25 12.5   2.86  1.66
  1.29  1.07 15.   14.  ]
GBD : [ 3.25  3.2   3.75  3.5   3.4   3.3   5.    3.6   4.75  4.2   3.8   4.25
  4.5   5.5   3.1   4.    6.    4.1   3.9   5.75  5.25  4.33  3.35   nan
  6.5   6.25  3.15  4.4   3.7   4.6   7.    2.95  3.    7.5   3.65  3.45
  3.33  7.25  3.85  8.    6.75  8.5   2.9   2.8   2.85  2.75  2.35  2.4
  2.65  9.25  9.    6.35  3.05  2.55  1.95  1.85  2.1   1.7   2.05  1.8
  2.6   2.25  2.3   2.2   1.75  1.45  2.5   2.7   3.55  5.35  3.95  4.3
  4.05  8.25  2.15  5.85  1.67  2.    5.05  7.75  9.75 11.   10.    8.75
  9.5 ]
GBA : [ 4.    3.75  2.5   5.5   1.65  1.7   2.25  2.8  10.    2.35  4.25  2.85
  1.6   1.8   7.5   2.75  2.15  4.2   7.    3.5   1.95  3.2   8.5   2.1
  3.    1.75  8.    1.73  6.    5.25  3.1   2.9   4.4   5.   11.    4.6
  2.3   2.2   3.3   2.6   1.4   5.75  4.75  2.7   2.45 12.    6.5   2.
  4.5   1.67  4.33  3.7   1.55  9.     nan  3.6   1.83  2.05  3.15  1.72
  4.8   2.4   7.75  3.8   3.4   4.3  13.    1.85  1.18  1.45  2.65  1.9
  6.25  3.9   1.5   1.77  3.25  2.55  9.5   4.1   1.62  1.57 15.    3.85
  1.53 16.    1.33  9.25  1.87  6.75  7.25  1.88  2.95 17.    2.38  1.48
  1.3  14.    1.42  1.36  2.63  3.35  1.46  1.44  1.37 12.5   1.28  1.27
 19.    1.35 21.    4.9   1.25 10.5   1.91 13.5   8.25  9.75 11.5   1.78
  8.75  4.7   3.55  1.47  4.15 15.5  20.   17.5  18.5  14.5   5.6   1.63
  5.2   1.2   1.34 18.    1.22  3.45  1.12  6.15  6.35  3.95  5.85  4.55
  3.65  7.85  4.65  7.9   3.05  6.9   7.7  19.5   5.35  1.13  1.15  5.05
  6.8   1.93  1.38  1.17 26.    1.23 25.    1.26  1.32 16.5   2.86  6.2
 24.   34.    1.52 23.   29.  ]
BSH : [ 1.73  1.91  2.3   1.44  4.75   nan  2.1   2.88  2.25  1.29  1.25  2.62
  1.83  2.75  1.7   6.    3.8   1.3   3.1   1.36  1.8   2.    2.5   3.5
  2.2   4.33  4.2   1.5   1.6   1.33  4.5   5.    2.38  1.57  1.2   1.67
  3.3   2.7   7.    1.53  2.4   1.14  1.62  2.8   1.4   3.    1.75  1.17
  1.18  5.5   1.95  1.22  3.4   9.    3.6   2.15  2.6   6.5   2.05  3.25
  3.2   2.37  4.    3.75  5.4   1.28  1.12  8.    1.61  2.87  8.5   7.5
 10.    1.85  1.65 11.    6.75  2.9  12.    5.25  1.9   6.25  1.1   1.11
  3.9  13.    9.5   2.45  2.17  2.63 17.    1.55 15.    5.75  1.08  1.06
  1.07  1.04  1.05  1.45  1.97  2.55]
BSD : [ 3.4   3.25  3.2   3.75  3.3    nan  4.5   5.    3.6   4.2   3.1   3.5
  6.    4.    4.33  5.5   4.75  6.5   3.8   7.    5.4   5.25  3.7   3.
  3.9   6.75  7.5   5.75  8.    8.5   2.88  2.9   2.38  2.8   2.4   2.62
  2.75  1.91  2.1   1.67  2.    1.7   1.33  2.25  1.4   2.6   1.73  1.44
  2.5   2.7   2.15  2.2   9.   10.   13.   11.   12.    3.45  5.95  4.4 ]
BSA : [ 4.2   3.6   2.75  6.5   1.67   nan  3.1   2.2   2.8   9.    2.38  3.8
  2.3   4.33  1.5   1.83  8.5   2.7   2.1   4.5   7.    4.    3.3   2.5
  1.91  2.9   3.    3.4   1.73  6.    5.    2.88  7.5   1.62  2.62  5.5
 10.    4.75  2.    1.4   3.5  15.    2.25  2.15 11.    1.75 12.    2.6
  1.53  1.7   1.95  5.4   1.2   1.57  1.85 13.    2.4   1.44  3.2   8.
  3.75  2.05  2.37  1.61  1.8   3.25  2.87 17.    1.33  1.29 23.   19.
  1.36  9.5   5.75  5.25  3.7  21.    1.3   1.25  1.55 26.    1.22  3.9
  1.6   2.63 10.5   1.12  4.6   1.17 16.   29.   34.    1.14  1.18  2.55
  3.15]
```

In [35]:

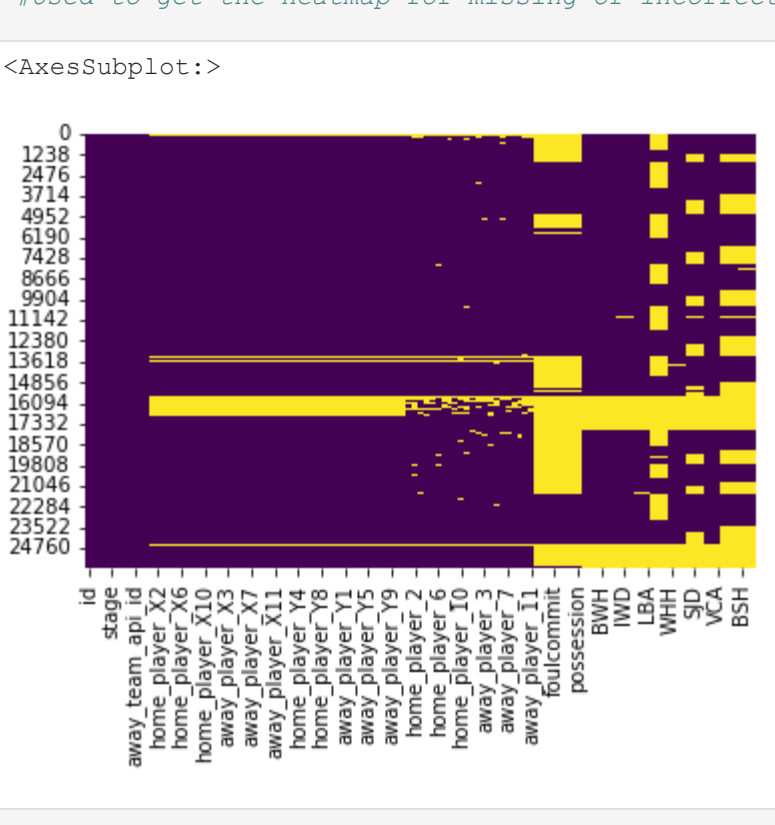
```
for col in df2.columns:
    df2[col].replace({'?':np.nan},inplace=True)
```

In [36]:

```
sns.heatmap(df2.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

Out[36]:

```
<AxesSubplot:>
```



In [63]:

```
num_col = []
for col in df2.columns:
    num_col.append(col)
```

In [64]:

```
for value in num_col:
    print(value)
```

```
id
country_id
league_id
season
stage
date
match_api_id
home_team_api_id
away_team_api_id
home_team_goal
away_team_goal
home_player_X1
home_player_X2
home_player_X3
home_player_X4
home_player_X5
home_player_X6
home_player_X7
home_player_X8
home_player_X9
home_player_X10
home_player_X11
away_player_X1
away_player_X2
away_player_X3
away_player_X4
away_player_X5
away_player_X6
away_player_X7
away_player_X8
away_player_X9
away_player_X10
away_player_X11
home_player_Y1
home_player_Y2
home_player_Y3
home_player_Y4
home_player_Y5
home_player_Y6
home_player_Y7
home_player_Y8
home_player_Y9
home_player_Y10
home_player_Y11
away_player_Y1
away_player_Y2
away_player_Y3
away_player_Y4
away_player_Y5
away_player_Y6
away_player_Y7
away_player_Y8
away_player_Y9
away_player_Y10
away_player_Y11
home_player_1
home_player_2
home_player_3
home_player_4
home_player_5
home_player_6
home_player_7
home_player_8
home_player_9
home_player_10
home_player_11
away_player_1
away_player_2
away_player_3
away_player_4
away_player_5
away_player_6
away_player_7
away_player_8
away_player_9
away_player_10
away_player_11
goal
shoton
shotoff
foulcommit
card
cross
corner
possession
B365H
B365D
B365A
BWH
BWD
BWA
IWH
IWD
IWA
LBH
LBD
LBA
PSH
PSD
PSA
WHH
WHD
WHA
SJH
SJD
SJA
VCH
VCD
VCA
GBH
GBD
GBA
BSH
BSD
BSA
```

In [65]:

```
#Replacing missing values with mean
num_col.remove('season')
num_col.remove('date')
num_col.remove('goal')
num_col.remove('shotoff')
num_col.remove('shoton')
num_col.remove('foulcommit')
num_col.remove('card')
num_col.remove('cross')
num_col.remove('corner')
num_col.remove('possession')
#We remove the above columns as we use analytical methods to deduce that missing values in the above columns ca
for col in num_col:
    df2[col]=pd.to_numeric(df2[col])
    df2[col].fillna(df2[col].mean(), inplace=True)
df2.head()
```

Out[65]:

	id	country_id	league_id	season	stage	date	match_api_id	home_team_api_id	away_team_api_id	home_team_goal	...	SJA	VCH	VC
0	1	1	1	2008/2009	1	17-08-2008 00:00	492473	9987	9993	1	...	4.00	1.65	3.4
1	2	1	1	2008/2009	1	16-08-2008 00:00	492474	10000	9994	0	...	3.80	2.00	3.2
2	3	1	1	2008/2009	1	16-08-2008 00:00	492475	9984	8635	0	...	2.50	2.35	3.2
3	4	1	1	2008/2009	1	17-08-2008 00:00	492476	9991	9998	5	...	7.50	1.45	3.7
4	5	1	1	2008/2009	1	16-08-2008 00:00	492477	7947	9985	1	...	1.73	4.50	3.4

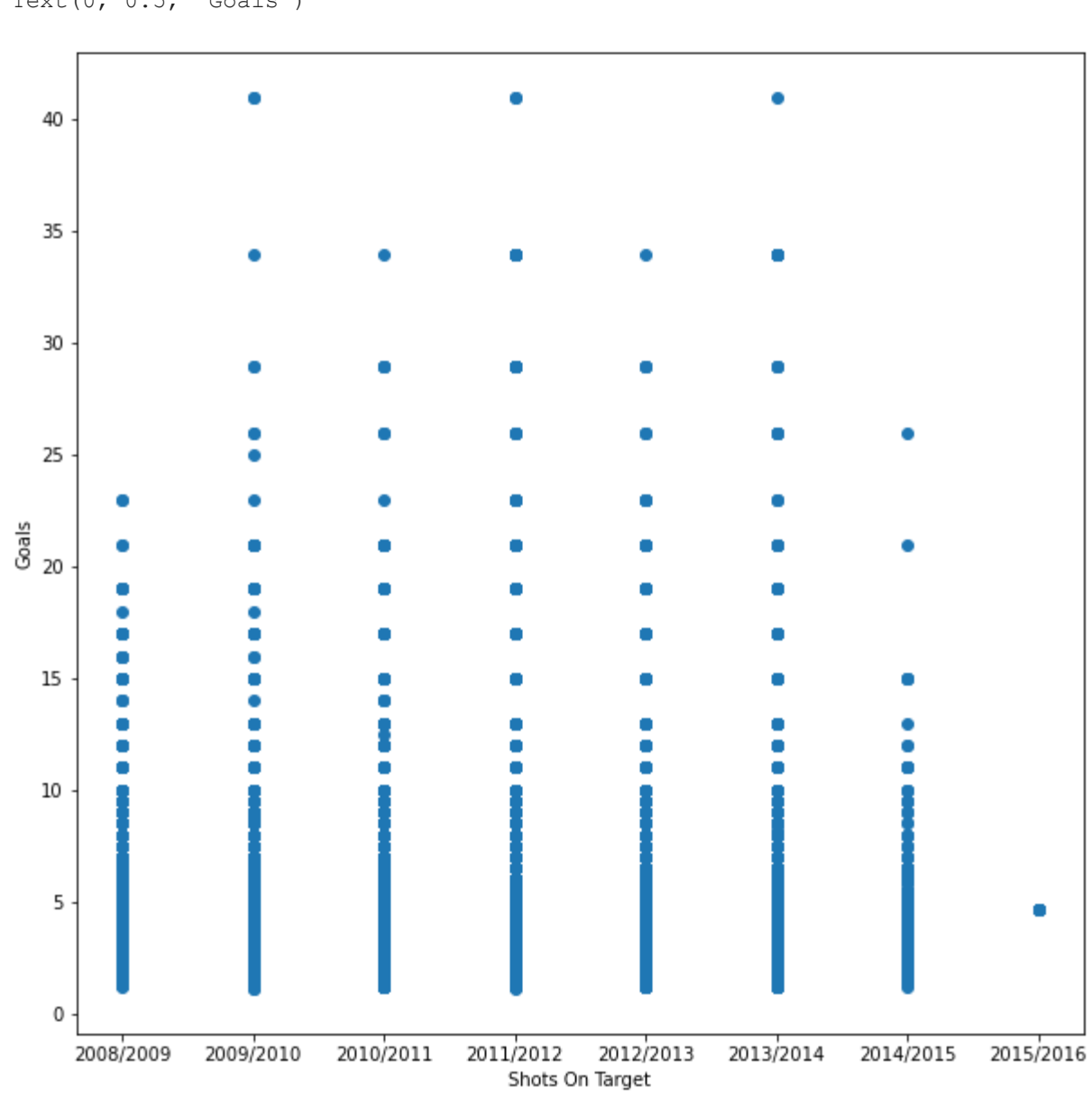
5 rows × 115 columns

In [68]:

```
#Testing correlations
plt.figure(figsize=(10,10))
plt.scatter(x='season',y='SJA',data=df2)
plt.xlabel('Season')
plt.ylabel('SJA')
```

Out[68]:

```
Text(0, 0.5, 'Goals')
```



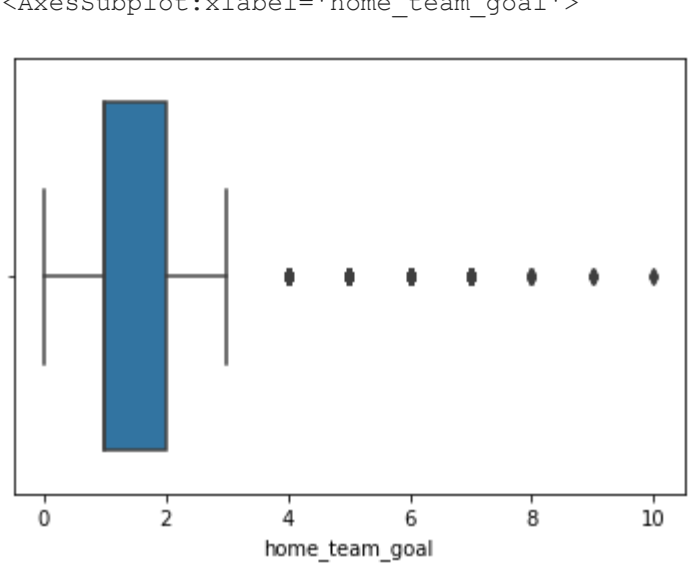
In [20]:

```
sns.boxplot(df2['home_team_goal'])
```

```
C:\Users\imsid\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variabl
e as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other
arguments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[20]:

```
<AxesSubplot:xlabel='home_team_goal'>
```



In [69]:

```
df3 = pd.read_csv('Player.csv')
```

In [70]:

```
df3.head()
```

Out[70]:

	id	player_api_id	player_name	player_fifa_api_id	birthday	height	weight
0	1	505942	Aaron Appindangoye	218353	1992-02-29 00:00:00	182.88	187
1	2	155782	Aaron Cresswell	189615	1989-12-15 00:00:00	170.18	146
2	3	162549	Aaron Doran	186170	1991-05-13 00:00:00	170.18	163
3	4	30572	Aaron Galindo	140161	1982-05-08 00:00:00	182.88	198
4	5	23780	Aaron Hughes	17725	1979-11-08 00:00:00	182.88	154

In [71]:

```
df3.describe()
```

Out[71]:

	id	player_api_id	player_fifa_api_id	height	weight
count	11060.000000	11060.000000	11060.000000	11060.000000	11060.000000
mean	5537.511392	156582.427215	165664.910488	181.867445	168.380289
std	3197.692647	160713.700624	58649.928360	6.369201	14.990217
min	1.000000	2625.000000	2.000000	157.480000	117.000000
25%	2767.750000	35555.500000	151889.500000	177.800000	159.000000
50%	5536.500000	96619.500000	184671.000000	182.880000	168.000000
75%	8306.250000	212470.500000	203883.250000	185.420000	179.000000
max	11075.000000	750584.000000	234141.000000	208.280000	243.000000

In [72]:

```
df3.shape
```

Out[72]:

```
(11060, 7)
```

In [73]:

```
df3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11060 entries, 0 to 11059
Data columns (total 7 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   id                  11060 non-null  int64
 1   player_api_id       11060 non-null  int64
 2   player_name         11060 non-null  object
 3   player_fifa_api_id  11060 non-null  int64
 4   birthday            11060 non-null  object
 5   height              11060 non-null  float64
 6   weight              11060 non-null  int64
dtypes: float64(1), int64(4), object(2)
memory usage: 605.0+ KB
```

In [82]:

```
#Checking for missing null values
df3.isnull().sum()
```

Out[82]:

```
id                    0
player_api_id         0
player_name           0
player_fifa_api_id    0
birthday              0
height                0
weight                0
dtype: int64
```

In [84]:

```
#Checking for wrong entries like symbols -,?,#,*,etc.
for col in df3.columns:
    print('{} : {}'.format(col,df3[col].unique()))
```

```
id : [    1     2     3 ... 11073 11074 11075]
player_api_id : [505942 155782 162549 ...  36491  35506  39902]
player_name : ['Aaron Appindangoye' 'Aaron Cresswell' 'Aaron Doran' ... 'Zsolt Low'
 'Zurab Khizanishvili' 'Zvjezdan Misimovic']
player_fifa_api_id : [218353 189615 186170 ... 111191  47058 102359]
birthday : ['1992-02-29 00:00:00' '1989-12-15 00:00:00' '1991-05-13 00:00:00' ...
 '1996-01-10 00:00:00' '1986-12-18 00:00:00' '1979-04-29 00:00:00']
height : [182.88 170.18 172.72 165.1  190.5  175.26 187.96 177.8  185.42 180.34
 200.66 198.12 193.04 167.64 195.58 162.56 203.2  160.02 157.48 208.28]
weight : [187 146 163 198 154 161 139 181 170 150 168 143 176 159 185 172 165 157
 179 174 212 218 190 148 152 141 132 201 183 205 194 137 203 196 126 192
 134 214 130 209 128 207 216 123 220 225 227 117 243 121]
```

In [86]:

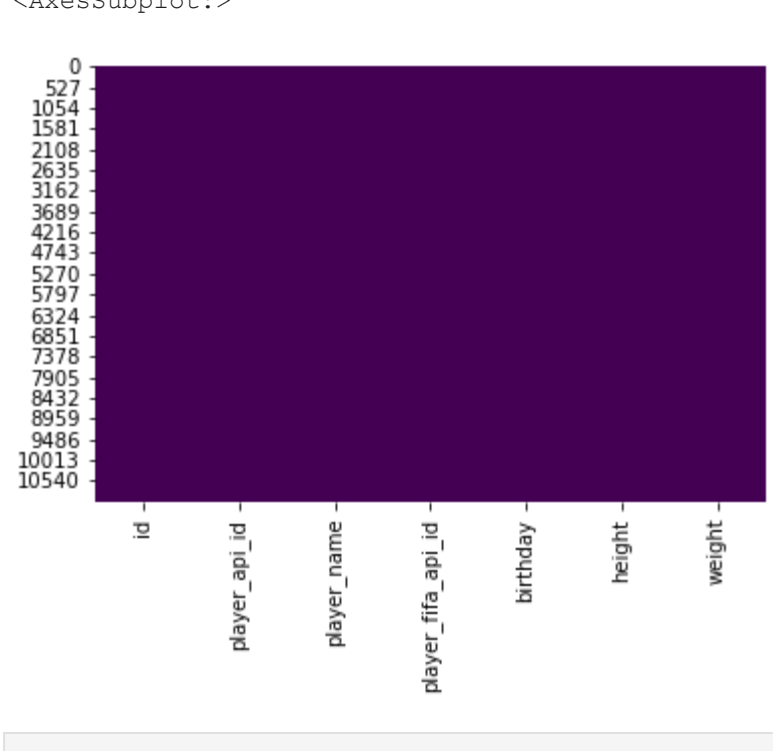
```
for col in df3.columns:
    df3[col].replace({'?':np.nan},inplace=True)
```

In [87]:

```
sns.heatmap(df3.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

Out[87]:

```
<AxesSubplot:>
```

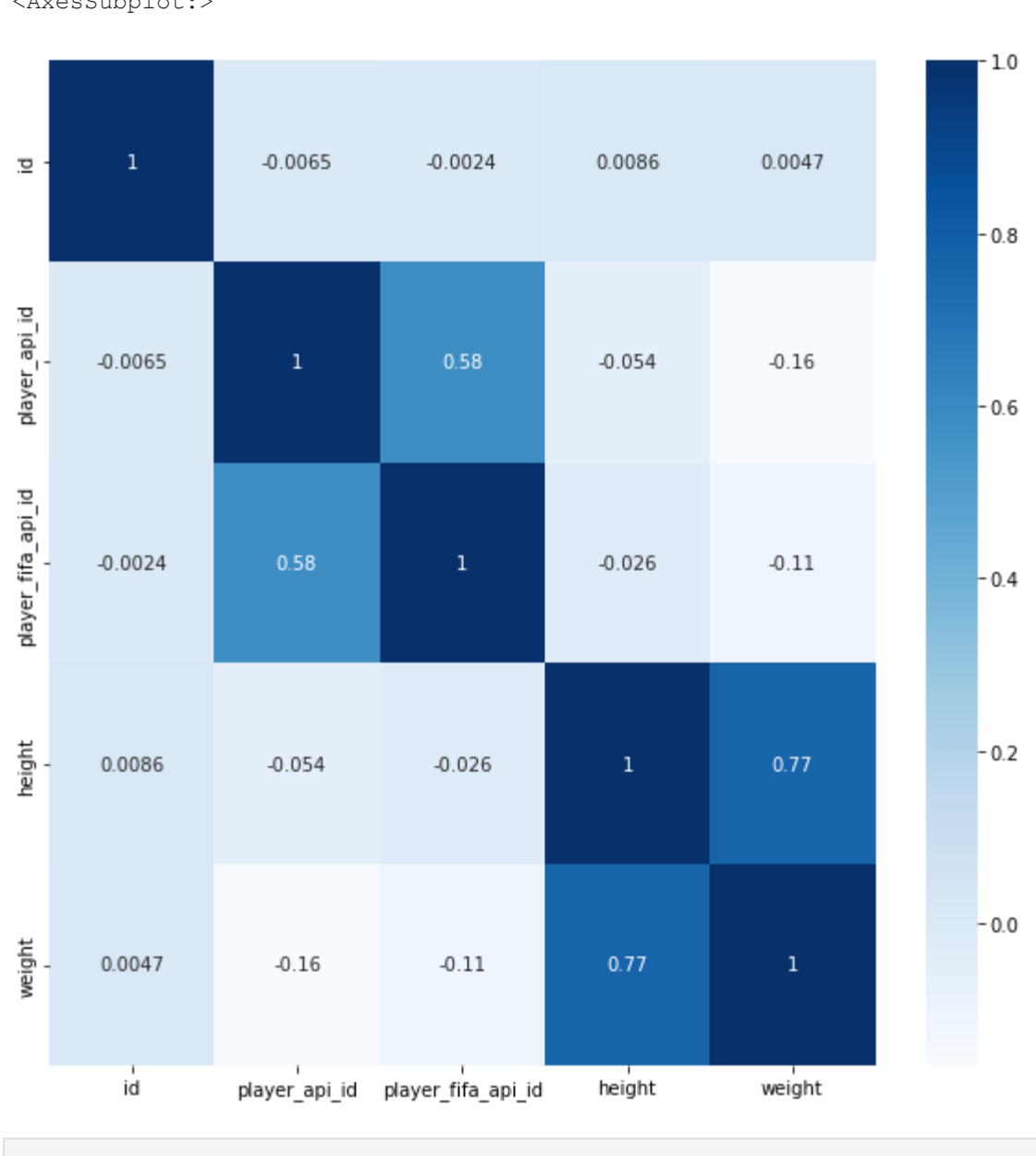


In [88]:

```
plt.figure(figsize=(10,10))
sns.heatmap(df3.corr(),cbar=True,annot=True,cmap='Blues')
```

Out[88]:

```
<AxesSubplot:>
```



In [91]:

```
df4 = pd.read_csv('Player_Attributes.csv')
```

In [92]:

```
df4.head()
```

Out[92]:

	id	player_fifa_api_id	player_api_id	date	overall_rating	potential	preferred_foot	attacking_work_rate	defensive_work_rate	crossing	...
0	1	218353	505942	2016-02-18 00:00:00	67.0	71.0	right	medium	medium	49.0	...
1	2	218353	505942	2015-11-19 00:00:00	67.0	71.0	right	medium	medium	49.0	...
2	3	218353	505942	2015-09-21 00:00:00	62.0	66.0	right	medium	medium	49.0	...
3	4	218353	505942	2015-03-20 00:00:00	61.0	65.0	right	medium	medium	48.0	...
4	5	218353	505942	2007-02-22 00:00:00	61.0	65.0	right	medium	medium	48.0	...

5 rows × 42 columns

In [93]:

```
df4.describe()
```

Out[93]:

	id	player_fifa_api_id	player_api_id	overall_rating	potential	crossing	finishing	heading_accuracy	short_pa
count	183978.00000	183978.000000	183978.000000	183142.000000	183142.000000	183142.000000	183142.000000	183142.000000	183142.0
mean	91989.50000	165671.524291	135900.617324	68.600015	73.460353	55.086883	49.921078	57.266023	62.4
std	53110.01825	53851.094769	136927.840510	7.041139	6.592271	17.242135	19.038705	16.488905	14.1
min	1.00000	2.000000	2625.000000	33.000000	39.000000	1.000000	1.000000	1.000000	3.0
25%	45995.25000	155798.000000	34763.000000	64.000000	69.000000	45.000000	34.000000	49.000000	57.0
50%	91989.50000	183488.000000	77741.000000	69.000000	74.000000	59.000000	53.000000	60.000000	65.0
75%	137983.75000	199848.000000	191080.000000	73.000000	78.000000	68.000000	65.000000	68.000000	72.0
max	183978.00000	234141.000000	750584.000000	94.000000	97.000000	95.000000	97.000000	98.000000	97.0

8 rows × 38 columns

In [94]:

```
df4.shape
```

Out[94]:

```
(183978, 42)
```

In [95]:

```
df4.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 183978 entries, 0 to 183977
Data columns (total 42 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   id                   183978 non-null  int64
 1   player_fifa_api_id   183978 non-null  int64
 2   player_api_id        183978 non-null  int64
 3   date                 183978 non-null  object
 4   overall_rating       183142 non-null  float64
 5   potential            183142 non-null  float64
 6   preferred_foot       183142 non-null  object
 7   attacking_work_rate  180748 non-null  object
 8   defensive_work_rate  183142 non-null  object
 9   crossing             183142 non-null  float64
 10  finishing            183142 non-null  float64
 11  heading_accuracy     183142 non-null  float64
 12  short_passing        183142 non-null  float64
 13  volleys              181265 non-null  float64
 14  dribbling            183142 non-null  float64
 15  curve                181265 non-null  float64
 16  free_kick_accuracy   183142 non-null  float64
 17  long_passing         183142 non-null  float64
 18  ball_control         183142 non-null  float64
 19  acceleration         183142 non-null  float64
 20  sprint_speed         183142 non-null  float64
 21  agility              181265 non-null  float64
 22  reactions            183142 non-null  float64
 23  balance              181265 non-null  float64
 24  shot_power           183142 non-null  float64
 25  jumping              181265 non-null  float64
 26  stamina              183142 non-null  float64
 27  strength             183142 non-null  float64
 28  long_shots           183142 non-null  float64
 29  aggression           183142 non-null  float64
 30  interceptions        183142 non-null  float64
 31  positioning          183142 non-null  float64
 32  vision               181265 non-null  float64
 33  penalties            183142 non-null  float64
 34  marking              183142 non-null  float64
 35  standing_tackle      183142 non-null  float64
 36  sliding_tackle       181265 non-null  float64
 37  gk_diving            183142 non-null  float64
 38  gk_handling          183142 non-null  float64
 39  gk_kicking           183142 non-null  float64
 40  gk_positioning       183142 non-null  float64
 41  gk_reflexes          183142 non-null  float64
dtypes: float64(35), int64(3), object(4)
memory usage: 59.0+ MB
```

In [100]:

```
#Checking for missing null values
df4.isnull().sum()
```

Out[100]:

```
id                        0
player_fifa_api_id        0
player_api_id             0
date                      0
overall_rating          836
potential               836
preferred_foot          836
attacking_work_rate    3230
defensive_work_rate     836
crossing                836
finishing               836
heading_accuracy        836
short_passing           836
volleys                2713
dribbling               836
curve                  2713
free_kick_accuracy      836
long_passing            836
ball_control            836
acceleration            836
sprint_speed           2713
agility                2713
reactions               836
balance                2713
shot_power              836
jumping                2713
stamina                 836
strength                836
long_shots              836
aggression              836
interceptions           836
positioning             836
vision                 2713
penalties               836
marking                 836
standing_tackle         836
sliding_tackle         2713
gk_diving               836
gk_handling             836
gk_kicking              836
gk_positioning          836
gk_reflexes             836
dtype: int64
```

In [99]:

```
#Checking for wrong entries like symbols -,?,#,*,etc.
for col in df4.columns:
    print('{} : {}'.format(col,df4[col].unique()))
```

```
id : [     1      2      3 ... 183976 183977 183978]
player_fifa_api_id : [218353 189615 186170 ... 111191  47058 102359]
player_api_id : [505942 155782 162549 ...  36491  35506  39902]
date : ['2016-02-18 00:00:00' '2015-11-19 00:00:00' '2015-09-21 00:00:00'
 '2015-03-20 00:00:00' '2007-02-22 00:00:00' '2016-04-21 00:00:00'
 '2016-04-07 00:00:00' '2016-01-07 00:00:00' '2015-12-24 00:00:00'
 '2015-12-17 00:00:00' '2015-10-16 00:00:00' '2015-09-25 00:00:00'
 '2015-01-09 00:00:00' '2015-10-09 00:00:00' '2014-11-07 00:00:00'
 '2014-09-18 00:00:00' '2014-05-09 00:00:00' '2014-04-04 00:00:00'
 '2014-02-14 00:00:00' '2013-03-08 00:00:00' '2013-11-08 00:00:00'
 '2013-10-04 00:00:00' '2013-05-10 00:00:00' '2013-05-03 00:00:00'
 '2013-03-22 00:00:00' '2013-03-15 00:00:00' '2013-02-22 00:00:00'
 '2013-02-15 00:00:00' '2012-08-31 00:00:00' '2012-02-22 00:00:00'
 '2011-08-30 00:00:00' '2011-02-22 00:00:00' '2010-02-22 00:00:00'
 '2010-08-30 00:00:00' '2009-02-22 00:00:00' '2009-08-30 00:00:00'
 '2008-08-30 00:00:00' '2014-09-19 00:00:00' '2014-04-18 00:00:00'
 '2014-01-31 00:00:00' '2013-09-20 00:00:00' '2013-05-31 00:00:00'
 '2014-05-02 00:00:00' '2015-02-20 00:00:00' '2013-04-05 00:00:00'
 '2014-03-28 00:00:00' '2013-09-27 00:00:00' '2015-10-02 00:00:00'
 '2015-07-03 00:00:00' '2013-02-01 00:00:00' '2015-01-16 00:00:00'
 '2014-11-28 00:00:00' '2015-03-13 00:00:00' '2014-04-11 00:00:00'
 '2016-05-05 00:00:00' '2014-03-14 00:00:00' '2015-05-08 00:00:00'
 '2014-10-17 00:00:00' '2014-09-12 00:00:00' '2016-04-28 00:00:00'
 '2016-03-10 00:00:00' '2014-09-05 00:00:00' '2014-02-28 00:00:00'
 '2016-02-25 00:00:00' '2015-10-23 00:00:00' '2016-02-04 00:00:00'
 '2015-05-15 00:00:00' '2013-12-06 00:00:00' '2015-02-06 00:00:00'
 '2015-01-23 00:00:00' '2015-10-30 00:00:00' '2013-11-01 00:00:00'
 '2015-03-06 00:00:00' '2014-11-14 00:00:00' '2014-12-27 00:00:00'
 '2015-04-17 00:00:00' '2016-03-17 00:00:00' '2015-12-03 00:00:00'
 '2015-03-27 00:00:00' '2014-02-07 00:00:00' '2015-10-23 00:00:00'
 '2015-02-13 00:00:00' '2014-11-21 00:00:00' '2015-11-06 00:00:00'
 '2014-10-24 00:00:00' '2014-10-10 00:00:00' '2015-02-13 00:00:00'
 '2015-11-26 00:00:00' '2015-10-30 00:00:00' '2013-11-22 00:00:00'
 '2016-01-28 00:00:00' '2016-03-24 00:00:00' '2016-04-14 00:00:00'
 '2013-09-13 00:00:00' '2015-05-15 00:00:00' '2013-12-20 00:00:00'
 '2015-01-16 00:00:00' '2015-05-01 00:00:00' '2014-12-19 00:00:00'
 '2013-11-15 00:00:00' '2014-12-27 00:00:00' '2013-10-11 00:00:00'
 '2014-04-25 00:00:00' '2014-09-26 00:00:00' '2015-02-27 00:00:00'
 '2016-03-10 00:00:00' '2016-03-03 00:00:00' '2016-05-12 00:00:00'
 '2016-09-27 00:00:00' '2016-08-16 00:00:00' '2013-03-28 00:00:00'
 '2015-09-16 00:00:00' '2015-11-26 00:00:00' '2013-06-07 00:00:00'
 '2016-01-13 00:00:00' '2016-06-23 00:00:00' '2014-05-16 00:00:00'
 '2015-01-23 00:00:00' '2015-09-29 00:00:00' '2014-10-24 00:00:00'
 '2014-05-09 00:00:00' '2016-05-05 00:00:00' '2013-05-17 00:00:00'
 '2014-11-28 00:00:00' '2014-11-26 00:00:00' '2015-03-27 00:00:00'
 '2015-06-09 00:00:00' '2015-04-10 00:00:00' '2014-02-21 00:00:00'
 '2014-04-16 00:00:00' '2015-03-09 00:00:00' '2016-03-31 00:00:00'
 '2016-01-28 00:00:00' '2015-12-10 00:00:00' '2008-02-22 00:00:00'
 '2015-01-16 00:00:00' '2016-02-03 00:00:00' '2016-02-11 00:00:00'
 '2015-02-20 00:00:00' '2015-05-08 00:00:00' '2013-07-12 00:00:00'
 '2015-10-17 00:00:00' '2015-07-05 00:00:00' '2015-10-23 00:00:00'
 '2015-01-30 00:00:00' '2015-08-06 00:00:00' '2016-02-19 00:00:00'
 '2016-09-13 00:00:00' '2013-11-16 00:00:00' '2016-07-07 00:00:00'
 '2016-04-24 00:00:00' '2014-08-14 00:00:00' '2016-06-02 00:00:00'
 '2015-09-30 00:00:00' '2014-09-23 00:00:00' '2014-05-23 00:00:00'
 '2016-05-19 00:00:00' '2015-05-01 00:00:00' '2013-09-06 00:00:00'
 '2016-03-17 00:00:00' '2014-10-15 00:00:00' '2016-05-26 00:00:00'
 '2013-10-19 00:00:00' '2014-09-16 00:00:00' '2013-08-23 00:00:00'
 '2015-08-27 00:00:00' '2015-05-11 00:00:00' '2015-07-24 00:00:00'
 '2013-08-30 00:00:00' '2013-08-09 00:00:00' '2014-08-08 00:00:00'
 '2014-08-22 00:00:00' '2014-08-15 00:00:00' '2014-06-21 00:00:00'
 '2013-06-26 00:00:00' '2014-08-29 00:00:00' '2013-07-19 00:00:00'
 '2013-06-28 00:00:00' '2014-08-02 00:00:00' '2013-09-13 00:00:00'
 '2013-12-30 00:00:00' '2014-08-23 00:00:00' '2008-08-21 00:00:00'
 '2014-06-30 00:00:00' '2016-04-05 00:00:00' '2014-08-01 00:00:00'
 '2014-08-07 00:00:00' '2015-03-04 00:00:00' '2015-01-28 00:00:00'
 '2014-09-05 00:00:00' '2015-09-14 00:00:00' '2014-09-26 00:00:00'
 '2015-09-10 00:00:00' '2016-06-26 00:00:00' '2015-09-01 00:00:00'
 '2016-12-13 00:00:00' '2016-07-27 00:00:00']
overall_rating : [67. 62. 61. 74. 73. 71. 70. 69. 68. 65. 64. 54. 51. 52. 47. 53. 66. 59.
 75. 72. 76. 78. 77. 79. 60. 81. 80. 82. 84. 83. 63. 58. 55. 50. 56.
 nan 49. 57. 42. 46. 45. 85. 44. 86. 89. 87. 88. 91. 40. 90. 41. 43. 38.
 91. 92. 39. 33. 36. 37. 35. 94.]
potential : [71. 66. 65. 76. 75. 77. 78. 79. 80. 68. 74. 81. 83. 86. 84. 85. 82. 70. 72. 69. 60. 67. 62. 61. 73.
 89. 54. 88. 52. 93. 32. 33. 51. 46. 44. 50. 47. 45. 87. 96. 39. 48. 40. 91. 95. 94. 48. 49. 42.
 97. 96. 39.]
preferred_foot : ['right' 'left' nan]
attacking_work_rate : ['medium' 'high' nan 'low' 'None' 'le' 'norm' 'stoc' 'y']
defensive_work_rate : ['medium' 'high' 'low' '_0' nan '5' 'ean' 'o' '1' 'ormal' '7' '2' '8' '4'
 'tocky' '0' '3' '6' '9' 'es']
crossing : [49. 48. 80. 79. 78. 77. 74. 58. 57. 22. 26. 70. 66. 40. 53. 55. 54. 59. 60. 46. 75. 76. 47. 25. 72. 71. 62. 52. 56. 73. 61.
 61. 39. 41. 42. 68. nan 35. 34. 50. 38. 84. 81. 82. 83. 64. 33. 29. 37. 19. 18. 83. 8. 31.
 36. 44. 43. 31. 32. 33. 15. 14. 9. 21. 28. 20. 27. 85. 16. 91. 93. 3. 89. 7.
 24. 91. 30. 10. 27. 86. 90. 88. 13. 17. 85. 87. 16. 5. 4. 94. 95. 2. 1.]
finishing : [44. 43. 53. 52. 51. 50. 40. 39. 48. 58. 56. 61. 69. 21. 20. 34. 33. 57.
 72. 73. 74. 71. 77. 79. 32. 31. 66. 65. 67. 68. 70. 62. 64. 15. 42. 41.
 45. 63. 54. 23. 75. 76. 26. 55. 29. 17. 16. 22. 60. 37. nan 30. 49.
 47. 46. 38. 28. 27. 80. 78. 11. 10. 19. 59. 36. 82. 13. 9. 5. 8. 35.
 14. 24. 81. 88. 92. 83. 84. 86. 90. 89. 93. 85. 19. 6. 3. 12. 4. 87.
 91. 7. 2. 1. 95. 94. 97. 96.]
heading_accuracy : [71. 70. 57. 58. 57. 56. 51. 50. 52. 62. 60. 59. 62. 45. 68. 67. 73. 74. 79. 72.
 76. 64. 65. 66. 47. 46. 28. 27. 48. 49. 54. 53. 55. 61. 80. 82. 43.
 78. 77. 75. 48. 69. 39. 30. 80. nan 81. 84. 13. 11. 25. 31. 23. 18. 12. 33. 18.
 37. 32. 90. 41. 10. 21. 35. 24. 18. 17. 33. 31. 91. 95. 94. 91. 97. 71. 1.
 83. 22. 19. 34. 88. 89. 15. 8. 92. 20. 14. 4. 91.
 7. 95. 5. 1. 94. 2. 98.]
short_passing : [61. 60. 71. 70. 69. 67. 64. 62. 59. 54. 79. 80. 75. 77. 76. 78. 73. 76. 72. 73. 74. 56. 57. 58. 68.
 55. 56. 40. 51. 49. nan 43. 47. 48. 34. 33. 41. 35. 15. 21. 33. 32. 28. 27. 50. 36. 37. 20. 30. 9. 13.
 12. 86. 87. 89. 18. 88. 91. 90. 92. 26. 1. 95. 17. 96. 7. 6. 5. 4.]
volleys : [44. 43. 40. 32. 29. 28. 30. 52. 51. 56. 63. 67. 69. 14. 60. 59. 65. 64. 27. 57. 25. 61. 79. 81. 78. 77. 76. 73.
 41. 24. 55. 42. 36. 22. 71. nan 58. 48. 45. 59. 61. 60. 58. 30. 50.
 53. 54. 35. 75. 71. 10. 11. 5. 22. 21. 66. 38. 18. 15. 19. 17. 9. 26. 80. 37. 20. 33. 13. 8. 31. 85. 82. 89. 84. 86.
 18. 17. 2. 88. 8. 6. 7. 90. 83. 4. 16. 1. 86. 91. 93. 95. 3. 1. 92. 93.
 91. 1. 93. 92.]
dribbling : [51. 50. 73. 71. 67. 66. 65. 64. 62. 44. 35. 56. 63. 61. 59. 60. 52. 57. 58. 53.
 59. 60. 52. 49. 58. 55. 54. 70. 79. 81. 80. 82.
 83. 86. 84. 85. 68. 75. 76. 77. 74. 72. 69. 78. 90. 89. 88. 87. 38. 39. 46. 36. 29. 24. 13. 29. 21. 25. 20. 26. 34.
 29. 19. 21. 16. 9. 23. 40. 12. 18. 12. 33. 24. 45. 19. 48. 10. 15. 47. 31. 37. 43. 27. 11. 14. 97. 92. 93. 95.
 22. 8. 96. 17. 94. 3. 5. 7. 95. 4. 2. 1. 91.]
curve : [45. 44. 70. 68. 67. 66. 61. 71. 72. 74. 73. 42. 43. 58. 59. 55. 56. 57. 53. 48. 52. 62. 61. 76. 75. 80. 79. 77. 78.
 59. 61. 60. 54. 25. 64. 69. 73. 38. 37. 35. 34. 47. 40. 48. 55. 56. 50. 51. 58. 41. 30. 39. 22. 26. 28. 29. 36. 12. 30. 25. 32.
 22. nan 46. 40. 26. 36. 33. 20. 18. 31. 27. 32. 23. 92. 88. 90. 85. 84. 82. 81. 83. 86. 87. 89. 91. 94. 93. 20. 24. 17.
 10. 7. 15. 95. 19. 1. 11. 16. 19. 18. 1. 16. 19.]
free_kick_accuracy : [39. 38. 69. 68. 43. 42. 26. 63. 61. 58. 59. 76. 77. 55. 70. 66. 64. 56. 67. 72. 65. 78. 79. 79. 77. 75. 74. 73.
 49. 47. 28. 29. 40. 41. 65. 52. 57. 62. 71. 14. 13. 56. 88. 89. 52. 30. 34.
 76. 86. 90. 85. 87. 86. 93. 94. 95. 96. 4. 2. 1.]
long_passing : [64. 63. 68. 67. 66. 65. 45. 49. 62. 62. 59. 38. 69. 72. 75. 79. 74. 71. 66. 56. 64. 60. 73.
 47. 39. 38. 33. 42. 27. 76. 25. 26. 21. 20. 53. 41. 40. 36. 21. 80. 82. 81. 78. 77. 83. 24. 13. 85. 86. 87.
 29. 16. 18. 12. 11. 35. 19. 15. 10. 90. 89. 92. 94. 95. 88. 93. 91.
 91. 6. 9. 93. 4. 3.]
ball_control : [49. 48. 71. 70. 68. 67. 66. 58. 59. 74. 63. 61. 64. 80. 79. 78. 77. 76. 75. 72. 69. 81. 82. 83. 84. 62. 69. 58. 55. 57.
 54. 41. 49. 46. 45. nan 57. 53. 50. 52. 47. 44. 43. 42. 28. 24. 27. 26. 22. 23. 21. 20. 19. 18. 32. 33.
 27. 26. 87. 45. 30. 32. 89. 40. 36. 37. 92. 90. 91. 86. 88. 85. 34. 93. 94. 14. 31. 95. 94. 25. 17. 14. 1. 96.
 7. 97. 8. 5.]
acceleration : [60. 79. 80. 84. 85. 69. 43. 42. 66. 67. 33. 37. 50. 55. 75. 73. 49. 54. 52. 53. 70. 72. 89.
 95. 15. 63. 47. 64. 31. 34. 61. 74. 58. 93. 87. 86. 36. 47. 87. 88. 44. 59. 91. 41. 35. 46. 39. 32. 48.
 28. 27. 22. 24. 29. 26. 23. 18. 38. 21. 13. 97. 16. 17. 10.
 21. 20. 23. 19. 96. 11. 94. 19. 10. 7. 9. 3. 12. 25. 5. 6. 1. 11. 8. 65. 4.]
sprint_speed : [64. 78. 82. 81. 72. 47. 46. 59. 74. 67. 57. 61. 75. 53. 54. 71. 69. 87. 91. 93.
 66. 68. 76. 63. 60. 85. 78. 70. 86. nan 80. 88. 73. 89. 90. 92. 95. 38. 38. 96. 45. 51. 50. 32. 26. 23. 29. 31. 33. 35.
 28. 27. 22. 20. 25. 19. 94. 97. 15. 6. 5. 11. 44. 41. 21. 30. 44. 41. 24. 12. 17. 14. 16. 18. 5. 4. 1. 2. 9. 10. 7. 3. 13. 8.]
agility : [59. 78. 72. 67. 55. 71. 73. 74. 85. 76. 66. 63. 71. 69. 69. 88.
 65. 50. 51. nan 56. 64. 47. 83. 34. 53. 67. 79. 84. 89. 58. 52. 68.
 61. 86. 62. 54. 41. 40. 38. 39. 92. 37. 22. 89. 28. 24. 21. 29. 16. 14. 27. 15. 19. 16. 97. 21.
 77. 75. 81. 82. 87. 80. 90. 91. 93. 94. 95. 96. 8. 7.]
reactions : [47. 46. 57. 76. 75. 60. 66. 70. 62. 61. 65. 79. 81. 78. 74. 71. 73. 77. 60. 84. 85. 86. 87. 88. 89. 90. 91. 92.
 52. 40. 22. 35. 44. 42. 33. 45. 39. 34. 36. 38. 37. 35. 25.
 21. 93. 94. 25. 95. 17.]
balance : [65. 90. 87. 62. 92. 84. 56. 41. 44. 45. 52. 94. 24. 78. 77. 71. 66. 53. 82. 75. 74. 70. 69. 60. 61. 54. 73. 80. 49.
 58. 59. 64. 68. 43. 50. 63. 81. 55. 47. 51. 48. 40. 30. 37. 39. 26. 33. 42. 39. 34. 39. 91. 35. 37. 23. 21. 22. 23. 26. 27.
 89. 93. 94. 96. 38. 28. 22. 14. 11. 18. 91. 13. 15. 95. 5. 4. 7. 2. 95. 1.]
shot_power : [55. 54. 71. 70. 65. 48. 47. 61. 65. 66. 60. 61. 54. 73. 66. 72. 56. 57. 62. 69. 76. 77. 76. 72. 67. 63. 60.
 73. 53. 52. 51. 62. 68. 63. 76. 74. 59. 58. 41. 46. 49. 34. 50. 38. 40. 59. 44. 53. 30. 20. 36. 32. 31. 35.
 24. 25. 27. 28. 15. 18. 89. 90. 88. 87. 85. 84. 86. 91. 93. 92. 94. 95. 96. 2. 4. 3. 1. 6. 8. 5. 7.]
jumping : [58. 57. 66. 67. 81. 79. 64. 68. 65. 71. 56. 70. 63. 76. 72. 62. 66. 61. 78. 46. 60. 54. nan 74. 71. 80. 84.
 35. 50. 53. 69. 59. 50. 86. 87. 49. 82. 47. 86. 36. 42. 45. 39. 94. 41. 92. 93. 32. 36. 30. 46. 33. 24. 31.
 26. 29. 28. 19. 20. 21. 23. 22.]
stamina : [54. 79. 80. 77. 70. 74. 73. 63. 59. 55. 56. 68. 75. 60. 62. 81. 82. 87. 71. 70. 89. 93.
 58. 44. 65. 40. 67. 85. 86. 84. 83. 88. 47. 55. 52. 42. 43. 50. 49. 51. 57. 46. 45. 53. 27. 30. 37. 38. 39. 25.
 19. 24. 14. 96. 26. 23. 95. 16. 20. 96. 15. 17. 10.]
strength : [76. 56. 50. 49. 48. 43. 64. 60. 69. 68. 61. 58. 72. 70. 79. 44. 72. 74. 59. 46. 37. 57. 78. 53.
 85. 89. 91. 94. 80. 91. 92. 94. 22. 21. 35. 93. 22.
 25. 26. 23. 24. 18. 92. 89. 92. 12. 10.]
long_shots : [35. 34. 62. 63. 60. 57. 58. 56. 55. 54. 59. 28. 33. 26. 25. 24. 73. 75. 78. 73. 52. 76. 79. 73. 89. 70.
 79. 70. 30. 23. 47. 46. 43. 41. 71. 39. 69. 50. 53. 40.
 42. 44. 64. 65. 53. 38. 61. 80. 82. 51. 46. 34. 31. 21. 22. 18. 90. 83. 84. 81. 19.
 84. 10. 93. 85. 92. 6. 9. 86. 91. 5. 4. 3. 7.
 7. 92. 1. 94. 95. 2. 96.]
aggression : [71. 63. 62. 68. 67. 66. 60. 59. 70. 51. 49. 36. 43. 44. 40. 28. 27. 75. 73. 61. 67. 40. 65.
 29. 30. 28. 76. 82. 78. 85. 94. 61. 78. 79. 80. 73. 43. 93.
 86. 50. 31. 37. 77. 23. 27. 23. 92. 88. 87. 26. 33. 22. 41. 16. 20. 21. 91. 32. 13. 11. 10.
 42. 22. 29. 20. 90. 97. 19. 25. 19. 24. 26. 25. 92. 24. 95. 96. 9. 12. 2. 1.
 95. 9.]
interceptions : [70. 41. 40. 67. 66. 65. 62. 59. 31. 32. 55. 54. 44. 35. 61. 68. 71. 77.
 80. 81. 83. 74. 56. 57. 36. 60. 73. 19. 64. 63. 58. 45. 43. 39. 69.
 72. 82. 30. 29. 24. 23. 51. 49. 48. 21. 27. 20. 25. 33. 28. 47. 50. 79.
 nan 52. 26. 76. 75. 42. 46. 84. 53. 12. 78. 34. 13. 38. 22. 15. 16. 10.
 18. 85. 11. 14. 17. 87. 7. 88. 93. 86. 92. 90. 9. 91. 8. 89. 95. 96.
 94. 6. 2. 5. 1. 3. 4.]
positioning : [45. 44. 60. 58. 57. 56. 19. 27. 28. 64. 67. 68. 66. 62. 50. 30. 26. 25.
 78. 81. 29. 76. 77. 71. 74. 75. 73. 39. 79. 69. 14. 65. 63. 55. 21. 70.
 41. 83. 84. 82. 80. 61. 52. 51. 49. 48. 72. 38. 23. 47. 53. 36. 17. 46.
 31. 54. nan 24. 22. 20. 11. 10. 18. 42. 35. 32. 59. 40. 85. 37. 33. 43.
 13. 12. 16. 15. 34. 87. 88. 4. 3. 89. 91. 95. 86. 90. 93. 5. 8. 9.
 6. 94. 2. 92. 7. 96.]
vision : [54. 53. 66. 65. 64. 61. 58. 63. 47. 38. 68. 71. 72. 73. 67. 41. 40. 60.
 76. 80. 82. 79. 83. 75. 77. 81. 15. 70. 62. 78. 31. 55. 50. 49. 44. 36.
 43. 74. 52. 51. 46. nan 33. 32. 69. 30. 57. 42. 56. 39. 35. 24. 25. 59.
 45. 34. 29. 48. 21. 18. 17. 20. 23. 16. 27. 26. 22. 28. 85. 87. 37. 14.
 10. 84. 12. 9. 13. 86. 2. 89. 90. 7. 5. 6. 19. 8. 94. 95. 96. 92.
 11. 93. 88. 91. 4. 1. 3. 97.]
penalties : [48. 47. 59. 58. 35. 28. 29. 61. 60. 63. 64. 66. 36. 37. 45. 44. 81. 82.
 76. 72. 68. 67. 73. 53. 42. 41. 62. 84. 70. 74. 56. 55. 52. 34. 69. 65.
 75. 71. 79. 54. 40. 39. 50. 85. 80. 32. 31. 22. 46. 57. nan 49. 43. 51.
 77. 38. 20. 25. 33. 83. 21. 78. 18. 17. 27. 86. 30. 12. 88. 94. 7. 16.
 19. 26. 23. 15. 93. 24. 91. 90. 92. 14. 9. 11. 13. 6. 87. 95. 10. 89.
 8. 5. 3. 2. 96.]
marking : [65. 62. 76. 73. 72. 66. 64. 53. 52. 23. 22. 21. 67. 78. 80. 77. 75. 74.
 81. 79. 82. 69. 28. 36. 30. 51. 29. 25. 18. 15. 63. 58. 60. 68. 56. 54.
 59. 57. 35. 34. 46. 43. 42. 61. 32. 26. 70. 13. 20. 47. 45. 83. nan 55.
 19. 44. 38. 17. 71. 48. 37. 39. 24. 14. 33. 49. 50. 12. 9. 31. 40. 27.
 16. 5. 84. 85. 10. 41. 11. 7. 6. 1. 88. 86. 87. 94. 93. 8. 4. 90.
 89. 91. 92. 3. 2. 96.]
standing_tackle : [69. 66. 63. 75. 72. 68. 64. 60. 59. 45. 22. 21. 71. 76. 78. 74. 70. 73.
 79. 80. 83. 34. 37. 30. 27. 25. 58. 56. 33. 23. 29. 15. 65. 62. 61. 54.
 20. 31. 24. 77. 67. 53. 47. 46. 19. 26. 42. 41. 32. 36. 51. 50. 57. 39.
 nan 55. 14. 13. 44. 43. 35. 81. 48. 52. 28. 17. 16. 84. 82. 11. 10. 40.
 18. 9. 12. 86. 85. 87. 88. 49. 8. 38. 2. 89. 90. 93. 95. 7. 91. 5.
 92. 6. 4. 3. 94. 1.]
sliding_tackle : [69. 66. 78. 72. 67. 63. 24. 21. 68. 65. 70. 71. 74. 75. 76. 77. 35. 32.
 29. 57. 56. 34. 25. 14. 12. 61. 60. 62. 73. 51. 44. 43. 17. 54. 52. 37.
 36. 45. 64. 59. 19. 18. 23. 33. 31. 38. 22. 30. nan 40. 53. 42. 55. 27.
 13. 82. 84. 79. 58. 46. 15. 11. 20. 41. 81. 47. 49. 26. 16. 50. 48. 39.
 80. 28. 10. 9. 83. 85. 6. 5. 7. 2. 4. 89. 90. 86. 87. 8. 88. 92.
 3. 91. 93. 94. 95.]
gk_diving : [ 6. 5. 14. 13. 16. 15. 8. 7. 12. 9. 53. 11. 10. 24. 33. nan 2. 4.
 70. 66. 65. 58. 3. 60. 51. 46. 77. 73. 75. 72. 63. 55. 69. 68. 64. 67.
 74. 71. 1. 80. 79. 78. 56. 61. 62. 49. 76. 59. 57. 20. 81. 22. 21. 31.
 54. 88. 83. 82. 48. 84. 18. 35. 90. 85. 86. 45. 19. 87. 17. 43. 44. 28.
 23. 52. 39. 47. 91. 93. 50. 26. 34. 32. 92. 89. 25. 94. 41. 40. 42. 29.
 27. 30. 36. 37.]
gk_handling : [11. 10. 7. 6. 22. 21. 12. 5. 20. 9. 13. 25. 8. 41. 15. 14. 23. 30.
 24. nan 16. 63. 62. 59. 4. 3. 2. 61. 49. 44. 71. 66. 72. 67. 56. 46.
 60. 58. 65. 69. 68. 48. 1. 75. 76. 79. 82. 70. 74. 73. 64. 55. 57. 53.
 51. 78. 77. 52. 19. 45. 43. 26. 54. 50. 32. 81. 83. 42. 29. 80. 84. 17.
 47. 86. 88. 18. 28. 87. 85. 91. 35. 92. 93. 90. 37. 27. 33. 89. 39. 36.
 40.]
gk_kicking : [10. 9. 8. 45. 46. 49. 12. 11. 54. 13. 64. 65. 61. 58. 16. 15. 57. 55.
 7. 6. 63. 48. 14. 59. 39. 73. 5. 79. 75. 72. 67. 21. 38. 33. 47. 52.
 42. 62. 27. 71. 76. nan 35. 53. 43. 51. 69. 44. 60. 66. 3. 2. 56. 70.
 78. 68. 31. 29. 74. 20. 41. 82. 81. 84. 23. 4. 36. 32. 50. 28. 25. 83.
 86. 37. 77. 30. 80. 34. 24. 40. 26. 1. 95. 96. 97. 89. 90. 88. 18. 91.
 22. 19. 87. 85. 94. 92. 93. 17.]
gk_positioning : [ 8. 7. 9. 22. 21. 12. 11. 20. 10. 16. 15. 6. 14. 13. 25. 51. 31. 24.
 23. nan 70. 65. 64. 39. 5. 3. 2. 19. 18. 62. 58. 63. 71. 72. 68. 60.
 50. 57. 53. 54. 59. 69. 67. 66. 4. 75. 76. 74. 61. 55. 56. 79. 78. 77.
 82. 81. 73. 52. 30. 80. 47. 32. 49. 43. 48. 84. 42. 1. 44. 83. 17. 86.
 85. 87. 88. 45. 28. 46. 92. 38. 91. 93. 90. 94. 34. 41. 89. 96. 35. 27.
 37. 29. 33. 26. 36.]
gk_reflexes : [ 8. 7. 12. 11. 22. 13. 21. 10. 20. 6. 15. 14. 9. 25. 53. 30. 5. 24.
 23. nan 4. 16. 75. 74. 3. 2. 64. 60. 58. 55. 78. 76. 73. 72. 68. 54.
 69. 63. 62. 66. 56. 80. 86. 85. 84. 81. 79. 52. 70. 18. 67. 71. 65. 59.
 61. 57. 83. 77. 82. 38. 17. 1. 48. 33. 50. 49. 45. 89. 88. 51. 87. 43.
 47. 46. 27. 37. 92. 90. 41. 19. 29. 44. 42. 91. 35. 26. 40. 93. 94. 96.
 32. 34. 31.]
```

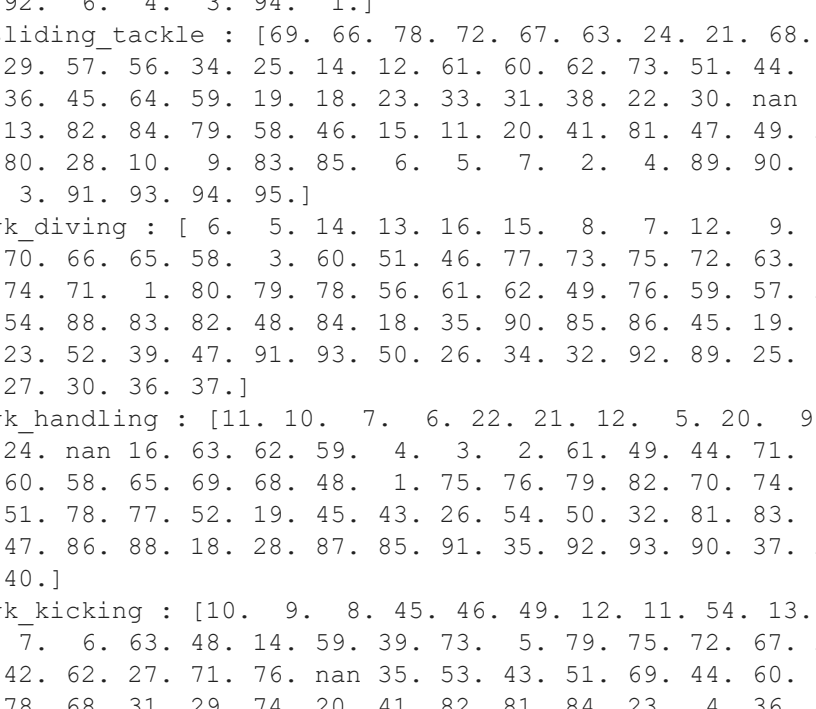
In [101]:

```
for col in df4.columns:
    df4[col].replace({'?':np.nan},inplace=True)
```

In [102]:

```
sns.heatmap(df4.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

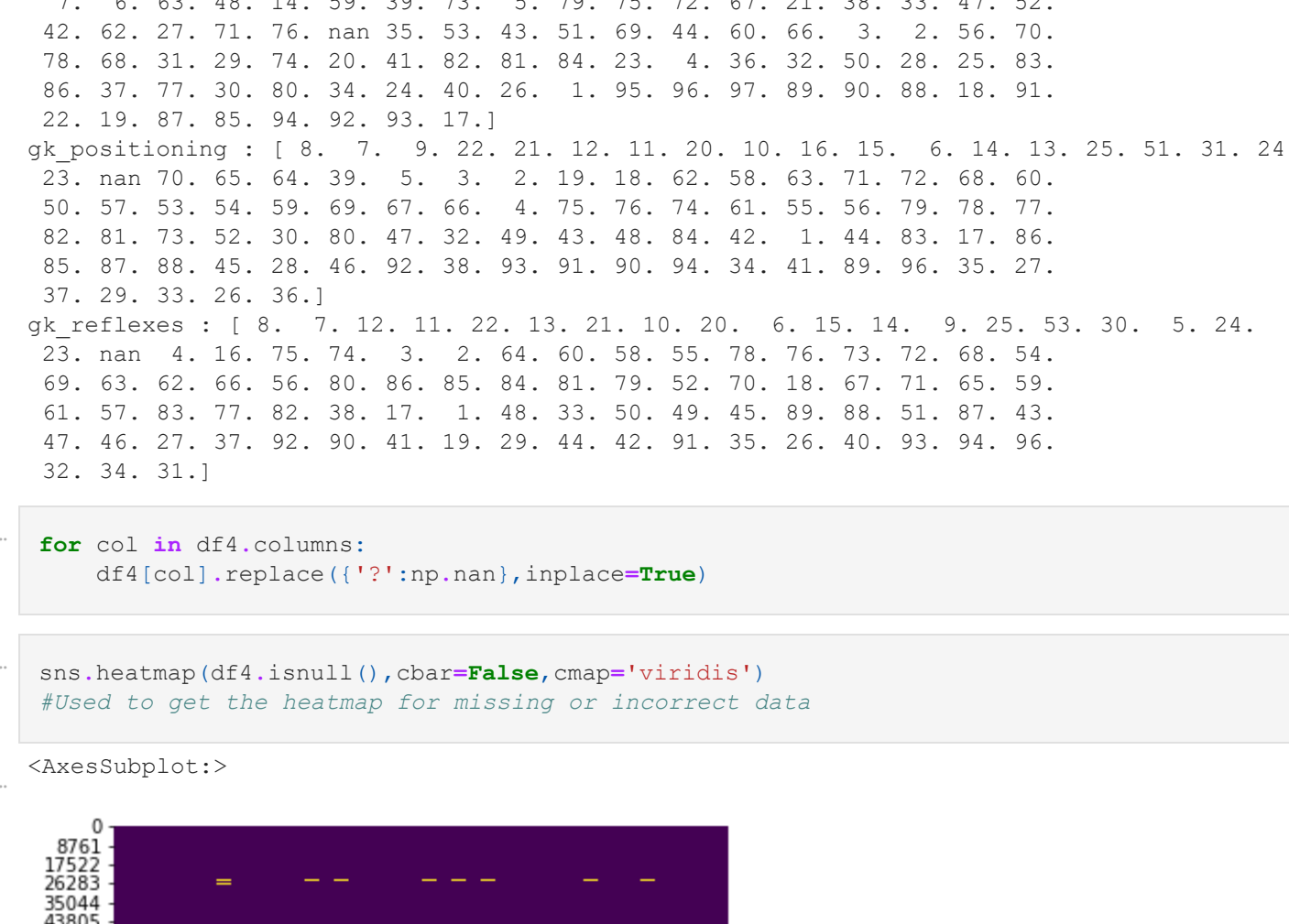
Out[102]: <AxesSubplot:>



In [103]:

```
#In previous heatmap we observe that missing data is negligible and can be dealt with later on
plt.figure(figsize=(10,10))
sns.heatmap(df4.corr(),cbar=True,annot=True,cmap='Blues')
```

Out[103]: <AxesSubplot:>



In [104]:

```
df5 = pd.read_csv('Team.csv')
```

In [105]:

```
df5.head()
```

Out[105]:

	id	team_api_id	team_fifa_api_id	team_long_name	team_short_name
0	1	9987	673.0	KRC Genk	GEN
1	2	9993	675.0	Beerschot AC	BAC
2	3	10000	15005.0	SV Zulte-Waregem	ZUL
3	4	9994	2007.0	Sporting Lokeren	LOK
4	5	9984	1750.0	KSV Cercle Brugge	CEB

In [106]:

```
df5.describe()
```

Out[106]:

	id	team_api_id	team_fifa_api_id
count	299.000000	299.000000	288.000000
mean	23735.301003	12340.521739	21534.305556
std	15167.914719	25940.411135	42456.439408
min	1.000000	1601.000000	1.000000
25%	9552.500000	8349.000000	178.750000
50%	22805.000000	8655.000000	673.500000
75%	36250.500000	9886.500000	1910.750000
max	51606.000000	274581.000000	112513.000000

In [107]:

```
df5.shape
```

Out[107]: (299, 5)

In [108]:

```
df5.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 299 entries, 0 to 298
Data columns (total 5 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   id                299 non-null    int64  
 1   team_api_id       299 non-null    int64  
 2   team_fifa_api_id  288 non-null    float64
 3   team_long_name    299 non-null    object 
 4   team_short_name   299 non-null    object 
dtypes: float64(1), int64(2), object(2)
memory usage: 11.8+ KB
```

In [110]:

```
#Checking for missing null values
df5.isnull().sum()
```

Out[110]:

```
id                   0
team_api_id          0
team_fifa_api_id    11
team_long_name       0
team_short_name      0
dtype: int64
```

In [112]:

```
#Checking for wrong entries like symbols -,?,#,*,etc.
for col in df5.columns:
    print('{} : {}'.format(col,df5[col].unique()))
```

```
id : [    1     2     3     4     5     6     7     8     9    10    11    12
    13    14    15    16    17    18   614  1034  1042  1513  2004  2476
  2510  3457  3458  3459  3460  3461  3462  3463  3464  3465  3466  3467
  3468  3469  3470  3471  3472  3473  3474  3475  3476  4218  4225  4234
  4996  5744  5747  5756  6504  6509  7261  7276  8021  8779  8784  9537
  9538  9539  9540  9541  9542  9543  9544  9545  9546  9547  9548  9549
  9550  9551  9552  9553  9554  9555  9556 10300 10309 10312 11074 11076
 11817 11822 11825 12587 12594 12595 13343 14106 14868 14876 15617 15618
 15619 15620 15621 15622 15623 15624 15625 15626 15627 15628 15629 15630
 15631 15632 15633 15634 16237 16239 16243 16848 16850 17465 18074 18079
 18691 19305 19916 19917 20513 20514 20515 20516 20517 20518 20519 20520
 20521 20522 20523 20524 20525 20526 20527 20528 20529 20530 20531 20532
 21280 21285 21292 22042 22044 22805 23523 24273 24288 25048 25791 25804
 26547 26548 26549 26550 26551 26552 26553 26554 26555 26556 26557 26558
 26559 26560 26561 26562 26563 26564 27161 27168 27780 29000 29617 29624
 30222 31443 31444 31445 31446 31447 31448 31449 31450 31451 31452 31453
 31454 31455 31456 31457 31458 31925 31928 32409 32891 33377 33855 34337
 34816 35283 35284 35285 35286 35287 35288 35289 35290 35291 35292 35293
 35294 35295 35296 35297 35298 35769 35774 36248 36253 36723 36735 37210
 37214 37696 38173 38176 38789 38791 39387 39388 39389 39390 39391 39392
 39393 39394 39395 39396 39397 39398 39853 40757 41216 41217 41673 43035
 43036 43037 43038 43039 43040 43041 43042 43043 43044 43045 43046 43047
 43048 43049 43050 43051 43052 43053 43054 43800 43803 43804 44557 44565
 44569 45330 45333 46087 46848 47605 47612 48358 49115 49116 49117 49118
 49119 49120 49121 49122 49123 49124 49479 49837 50201 50204 51606]
team_api_id : [  9987   9993  10000   9994   9984   8635   9991   9998   7947   9985
   8203   8342   9999   8571   4049   9996  10001   9986   9997   9989
   6351   1773   8475   8573 274581  10260  10261   9825   8659   8472
   8650   8654   8528  10252   8456   8668   8655   8549   8586   8559
  10194   8667   9879   8455   8462   8658   8602   8191   8483  10003
  10172   9850   8466   9798   9826   8344   8197   8678   9817   8583
   9830   9827   7819   9746   9831   8682   8689   9748   9941   9829
   9847   8481   8639   9851   8592   9874   9855   9873   9853   8588
  10249   4170 108893   8521   8576   4087   9836   9837   7794  10242
   9747   8550   8121   6391   9823   9790   8178   9789  10189   9904
   8721   8722   9810   8177   9912   8697   8398   8226   9788  10269
   8295   9911   8358   8165   9905   8350   8152   8406   8194   8357
   9776   8460   8234   8262   8524   8551   8529   8543   8530  10233
   8533   8690   8535   9885   8564   9857   8686   9875   9882   8636
   9804   9888   8600   8540   9976   8537  10167   9880   9858   6269
   9878   9876   7943   8534   9891 208931   8277   8674   9803   8611
   8525   8593   8464   8526   9908   8640   9791  10235   8614  10217
   6601  10228  10229   9761  10219   9839  10218   6413   7788   6433
   6631  10265   8031   8020   1601   8673   2183   8025   8030   8244
   8242   2182   8569   8322   1957   8028   2186   8245   8021   8024
   8033   8023   8027   8019 177361   9773   9807   9768   7992   7844
  10238   6403  10264  10213  10215   7841   9772   6421  10214   9809
  10212   9771   2033   9765  10211   4064   9764   8348   7842 158085
   6547   8613   6367 188163   8596   8548   9860   9927   8597  10251
   8485   8066   9925   9800   8429   9938   8467   8457   8284   8649
   8426  10267   8661   8371  10205   9783   8633   8388   8634   8696
   8302   9869   8305   8603   8479   8558  10281   8315   9865   9906
   9864   9868   8394   9867  10278   8581   8560   7878   8370   9910
  10268   8372   7869   8306   9956   6493  10192   9931   9930  10179
  10199   9824   7955  10243  10190  10191   9777   7730   7896]
team_fifa_api_id : [6.73000e+02 6.75000e+02 1.50050e+04 2.00700e+03 1.75000e+03 2.29000e+02
 6.74000e+02 1.74700e+03         nan 2.32000e+02 1.10724e+05 2.31000e+02
 5.46000e+02 1.00081e+05 1.11560e+05 6.81000e+02 6.70000e+02 6.80000e+02
 2.39000e+02 2.01300e+03 1.00087e+05 1.10913e+05 6.82000e+02 1.10000e+01
 1.30000e+01 1.00000e+00 1.09000e+02 1.06000e+02 9.00000e+00 1.90000e+01
 1.91700e+03 2.00000e+00 1.00000e+01 7.00000e+00 3.00000e+00 1.20000e+01
 1.80000e+01 4.00000e+00 1.80600e+03 1.95200e+03 1.44000e+02 5.00000e+00
 1.79000e+03 8.80000e+01 1.10000e+02 1.79600e+03 1.92600e+03 1.96000e+03
 1.50000e+01 1.79200e+03 1.70000e+01 1.79300e+03 1.79900e+03 1.96100e+03
 9.50000e+01 1.94300e+03 1.79500e+03 5.70000e+01 7.10000e+01 5.90000e+01
 2.10000e+02 1.73800e+03 7.20000e+01 1.73900e+03 2.17000e+02 6.60000e+01
 1.80900e+03 6.90000e+01 7.30000e+01 1.82300e+03 6.50000e+01 7.40000e+01
 2.19000e+02 2.26000e+02 1.80500e+03 1.10456e+05 1.81900e+03 6.40000e+01
 7.00000e+01 1.11376e+05 1.11989e+05 3.78000e+02 6.14000e+02 1.11271e+05
 1.10569e+05 3.79000e+02 5.80000e+01 2.94000e+02 6.20000e+01 6.80000e+01
 1.53000e+03 1.10316e+05 2.10000e+01 2.80000e+01 3.20000e+01 2.20000e+01
 3.40000e+01 4.85000e+02 1.75000e+02 3.10000e+01 1.82400e+03 1.66000e+02
 1.59000e+02 3.80000e+01 1.62000e+02 1.00290e+04 2.30000e+01 3.60000e+01
 1.83200e+03 1.60000e+02 2.50000e+01 1.71000e+02 1.69000e+02 2.90000e+01
 1.10329e+05 1.00409e+05 1.10636e+05 1.65000e+02 1.10500e+05 1.00300e+04
 1.11239e+05 1.10502e+05 3.90000e+01 1.83800e+03 1.84200e+03 4.60000e+01
 1.10364e+05 1.10556e+05 1.92000e+02 2.03000e+02 1.10374e+05 4.50000e+01
 4.70000e+01 1.89000e+02 5.20000e+01 4.80000e+01 1.83700e+03 4.40000e+01
 5.40000e+01 3.47000e+02 5.50000e+01 1.84300e+03 1.84800e+03 1.84400e+03
 5.00000e+01 1.10915e+05 1.90000e+02 1.12225e+05 2.00000e+02 2.06000e+02
 1.11974e+05 1.74600e+03 1.11657e+05 1.12409e+05 1.90900e+03 1.91500e+03
 1.90200e+03 1.90800e+03 1.90700e+03 2.45000e+02 1.91000e+03 6.35000e+02
 1.90300e+03 2.47000e+02 1.00624e+05 2.46000e+02 1.00646e+05 6.50000e+02
 1.91300e+03 1.90600e+03 1.90400e+03 1.90500e+03 1.00651e+05 1.97100e+03
 1.91400e+03 6.47000e+02 1.00632e+05 1.00626e+05 1.87300e+03 1.11429e+05
 8.74000e+02 1.87100e+03 1.57000e+03 1.11092e+05 1.11091e+05 3.01000e+02
 1.56400e+03 8.73000e+02 1.10744e+05 1.11082e+05 1.10745e+05 1.11086e+05
 1.10747e+05 1.11083e+05 1.10749e+05 1.11087e+05 1.10746e+05 1.12512e+05
 1.10565e+05 2.36000e+02 1.88900e+03 2.37000e+02 1.88700e+03 6.65000e+02
 1.89200e+03 1.89600e+03 1.90100e+03 7.44000e+02 2.34000e+02 1.00180e+04
 1.89100e+03 1.00737e+05 1.89300e+03 1.89500e+03 1.11540e+05 1.89700e+03
 1.88800e+03 1.90000e+03 1.00200e+04 1.12513e+05 1.00741e+05 1.89800e+03
 7.90000e+01 8.60000e+01 7.80000e+01 8.30000e+01 8.20000e+01 8.10000e+01
 7.70000e+01 6.20000e+02 7.80000e+01 1.00805e+05 1.84000e+02 1.81000e+02
 1.84000e+02 1.75400e+03 2.43000e+02 4.77000e+02 4.61000e+02 1.84100e+03
 1.86000e+03 1.76800e+03 2.61000e+02 4.49000e+02 1.11974e+05 2.06000e+02
 1.86100e+03 2.40000e+02 2.40000e+02 1.86100e+03 1.94800e+03 1.87700e+03
 1.00879e+05 1.77500e+03 1.86200e+03 1.10000e+05 1.85000e+03 1.10748e+05
 4.68000e+02 4.67000e+02 4.47000e+02 4.48000e+02 1.72000e+02 1.80100e+03
 9.00000e+00 8.96000e+02 8.94000e+02 2.39000e+02 2.29000e+02 1.80600e+03
 4.35000e+02 8.94000e+02 4.35000e+02]
team_long_name : ['KRC Genk' 'Beerschot AC' 'SV Zulte-Waregem' 'Sporting Lokeren'
 'KSV Cercle Brugge' 'RSC Anderlecht' 'KAA Gent' 'RAEC Mons'
 'FCV Dender EH' 'Standard de Liège' 'KV Mechelen' 'Club Brugge KV'
 'KSV Roeselare' 'KV Kortrijk' 'Tubize' 'Royal Excel Mouscron'
 'Oud-Heverlee Leuven' 'Waasland-Beveren' 'KVC Westerlo' 'Lierse SK'
 'KAS Eupen' 'Oud-Heverlee Leuven' 'Waasland-Beveren' 'KV Oostende'
 'Manchester United' 'Newcastle United' 'Arsenal' 'West Bromwich Albion'
 'Sunderland' 'Liverpool' 'West Ham United' 'Wigan Athletic' 'Aston Villa'
 'Manchester City' 'Everton' 'Blackburn Rovers' 'Middlesbrough'
 'Tottenham Hotspur' 'Bolton Wanderers' 'Stoke City' 'Hull City' 'Fulham'
 'Burnley' 'Portsmouth' 'Birmingham City' 'Wolverhampton Wanderers'
 'Burnley' 'Blackpool' 'Swansea City' 'Queens Park Rangers' 'Norwich City'
 'Southampton' 'Reading' 'Crystal Palace' 'Cardiff City' 'Leicester City'
 'Bournemouth' 'Watford' 'AJ Auxerre' 'FC Nantes' 'Girondins de Bordeaux'
 'SM Caen' 'Le Havre AC' 'OGC Nice' 'Le Mans FC' 'FC Lorient'
 'Olympique Lyonnais' 'Toulouse FC' 'AS Monaco' 'Paris Saint-Germain'
 'Olympique de Marseille' 'FC Sochaux-Montbéliard' 'Grenoble Foot 38'
 'Valenciennes FC' 'AS Saint-Étienne' 'RC Lens' 'Montpellier Hérault SC'
 'FC Lorient' 'Stade Brestois 29' 'Stade Rennais FC' 'AS Nancy-Lorraine'
 'LOSC Lille' 'Dijon FCO' 'Stade de Reims' 'AC Arles-Avignon'
 'AC Ajaccio' 'Évian Thonon Gaillard FC' 'Dijon FCO'
 'SC Bastia' 'ES Troyes AC' 'En Avant de Guingamp' 'FC Metz' 'Angers SCO'
 'GFC Ajaccio' 'FC Bayern Munich' 'Hamburger SV' 'Bayer 04 Leverkusen'
 '1. FC Köln' 'Eintracht Frankfurt' 'Hertha BSC Berlin' 'VfL Wolfsburg'
 'SV Werder Bremen' 'FC Schalke 04' 'Hannover 96' 'VfB Stuttgart'
 'TSG 1899 Hoffenheim' 'Borussia Dortmund' 'SC Freiburg' 'Energie Cottbus'
 '1. FSV Mainz 05' '1. FC Kaiserslautern' 'FC St. Pauli' 'FC Augsburg'
 'Fortuna Düsseldorf' 'SpVgg Greuther Fürth' 'Eintracht Braunschweig'
 'SC Paderborn 07' 'FC Ingolstadt 04' 'SV Darmstadt 98' 'Atalanta' 'Siena'
 'FC Bologna' 'Cagliari' 'Catania' 'Genoa' 'Chievo Verona' 'Reggio Calabria'
 'Fiorentina' 'Juventus' 'Milan' 'Bologna' 'Roma' 'Napoli' 'Sampdoria'
 'Inter' 'Torino' 'Lecce' 'Udinese' 'Palermo' 'Bari' 'Livorno' 'Parma'
 'Cesena' 'Brescia' 'Novara' 'Pescara' 'Hellas Verona' 'Sassuolo' 'Empoli'
 'Frosinone' 'Carpi' 'Vitesse' 'FC Groningen' 'Roda JC Kerkrade'
 'FC Twente' 'Willem II' 'Ajax' 'N.E.C.' 'De Graafschap' 'FC Utrecht'
 'PSV' 'Heracles Almelo' 'Feyenoord' 'Sparta Rotterdam' 'ADO Den Haag'
 'NAC Breda' 'NEC' 'SC Heerenveen' 'RKC Waalwijk' 'VVV-Venlo'
 'Excelsior' 'PEC Zwolle' 'SC Cambuur' 'Go Ahead Eagles' 'FC Dordrecht'
 'Wisła Kraków' 'Polonia Bytom' 'Ruch Chorzów' 'Legia Warszawa'
 'P. Warszawa' 'Śląsk Wrocław' 'Lechia Gdańsk' 'Widzew Łódź'
 'Odra Wodzisław' 'Lech Poznań' 'GKS Bełchatów' 'Arka Gdynia'
 'Jagiellonia Białystok' 'Piast Gliwice' 'Cracovia' 'Korona Kielce'
 'Zagłębie Lubin' 'Polonia Bytom' 'Pogoń Szczecin' 'Podbeskidzie Bielsko-Biała' 'Pogoń Szczecin'
 'Zawisza Bydgoszcz' 'Górnik Łęczna' 'Termalica Bruk-Bet Nieciecza'
 'FC Porto' 'CD Nacional' 'Rio Ave FC' 'Belenenses' 'SC Braga'
 'Vitória Guimarães' 'Vitória Setúbal' 'FC Paços de Ferreira' 'SC Braga'
 'CD Trofense' 'SL Benfica' 'Leixões SC' 'Académica de Coimbra'
 'Sporting CP' 'Estoril Praia' 'U.D. Leiria' 'União de Leiria, SAD'
 'S.C. Olhanense' 'Portimonense' 'SC Beira Mar' 'Feirense'
 'Gil Vicente FC' 'Moreirense FC' 'Marítimo' 'FC Arouca'
 'FC Penafiel' 'Boavista FC' 'Uniao da Madeira' 'Tondela' 'Falkirk'
 'Rangers' 'Heart of Midlothian' 'Motherwell' 'Kilmarnock' 'Hibernian'
 'Aberdeen' 'Inverness Caledonian Thistle' 'Celtic' 'St. Mirren'
 'Hamilton Academical FC' 'Dundee United' 'St. Johnstone FC'
 'Dunfermline Athletic' 'Dundee FC' 'Ross County FC'
 'Partick Thistle F.C.' 'Valencia CF' 'RCD Mallorca' 'CA Osasuna'
 'Villarreal CF' 'RC Deportivo de La Coruña' 'Real Madrid CF'
 'CD Numancia' 'FC Barcelona' 'Racing Santander' 'Sevilla FC'
 'Real Sporting de Gijón' 'Getafe CF' 'Real Betis Balompié'
 'RC Recreativo' 'RCD Espanyol' 'Real Valladolid'
 'Athletic Club de Bilbao' 'UD Almería' 'Atlético Madrid' 'Málaga CF'
 'Xerez Club Deportivo' 'Real Zaragoza' 'CD Tenerife'
 'Levante UD' 'Hércules Club de Fútbol' 'Real Sociedad' 'Granada CF'
 'Rayo Vallecano' 'RC Celta de Vigo' 'Elche CF' 'SD Eibar' 'Córdoba CF'
 'UD Las Palmas' 'Grasshopper Club Zürich' 'AC Bellinzona'
 'BSC Young Boys' 'FC Basel' 'FC Aarau' 'FC Sion' 'FC Luzern' 'FC Vaduz'
 'Neuchâtel Xamax' 'FC St. Gallen' 'FC Thun' 'Servette FC'
 'FC Lausanne-Sports' 'Lugano']
team_short_name : ['GEN' 'BAC' 'ZUL' 'LOK' 'CEB' 'AND' 'GEN' 'MON' 'DEN' 'STL' 'MEC' 'CLB' 'ROS'
 'KOR' 'TUB' 'MOU' 'WES' 'CHA' 'STT' 'LIE' 'EUP' 'O-H' 'WAA' 'OOS' 'MUN'
 'NEW' 'ARS' 'WBA' 'SUN' 'LIV' 'WHU' 'WIG' 'AVL' 'MCI' 'EVE' 'BLB'
 'MID' 'TOT' 'BOL' 'STK' 'HUL' 'FUL' 'CHE' 'POR' 'BIR' 'WOL' 'BUR' 'BLA'
 'SWA' 'QPR' 'NOR' 'SOU' 'REA' 'CRY' 'CAR' 'LEI' 'BOU' 'WAT' 'AUX' 'NAN'
 'BOR' 'CAE' 'LEH' 'NIC' 'LEM' 'LOR' 'LYO' 'TOU' 'ETG' 'DIJ' 'REI' 'BAS'
 'SOC' 'GRE' 'VAL' 'ETI' 'LEN' 'MON' 'NAN' 'AJA' 'GAZ' 'BAS' 'NIC' 'TRO'
 'GUI' 'MET' 'ANG' 'GAJ' 'BMU' 'HAM' 'LEV' 'DOR' 'EIN' 'HER' 'WOL' 'BRE'
 'SCH' 'HAN' 'STU' 'HOF' 'DOR' 'FRE' 'COT' 'MAI' 'KAI' 'STP' 'AUG' 'FOR'
 'GRF' 'BRA' 'PAD' 'ING' 'DAR' 'ATA' 'SIE' 'BOL' 'CAG' 'CAT' 'GEN' 'CHI'
 'REG' 'FIO' 'JUV' 'ACM' 'ROM' 'NAP' 'SAM' 'INT' 'TOR' 'LEC' 'UDI' 'PAL'
 'BAR' 'LIV' 'PAR' 'CES' 'BRE' 'NOV' 'PES' 'VER' 'SAS' 'EMP' 'FRO' 'CAR'
 'VIT' 'GRO' 'ROD' 'TWE' 'WII' 'AJA' 'NEC' 'GRA' 'UTR' 'PSV' 'HER' 'FEY'
 'SPA' 'ADO' 'NAC' 'NEC' 'HEE' 'RKC' 'VEN' 'EXC' 'ZWO' 'CAM' 'GAE' 'DOR'
 'WIS' 'POB' 'GOR' 'CHO' 'LEG' 'PWA' 'SLA' 'LGD' 'WID' 'ODR' 'POZ' 'BEL'
 'ARK' 'BIA' 'PIA' 'CRA' 'KKI' 'ZAG' 'POG' 'POD' 'POG' 'ZAW' 'TER' 'POR'
 'NAC' 'RIO' 'BEL' 'BRA' 'GUI' 'SET' 'FER' 'TRO' 'BEN' 'LEI' 'ACA' 'SCP'
 'EST' 'ULE' 'UDL' 'OLH' 'POR' 'BEI' 'FEI' 'GIL' 'MOR' 'MAR' 'ARO' 'PEN'
 'BOA' 'UMA' 'TON' 'FAL' 'RAN' 'HEA' 'MOT' 'KIL' 'HIB' 'ABE' 'INV' 'CEL'
 'MIR' 'HAM' 'DUN' 'STJ' 'DUN' 'DUN' 'ROS' 'PAR' 'VAL' 'MAL' 'OSA' 'VIL'
 'COR' 'REA' 'NUM' 'BAR' 'RAC' 'SEV' 'GIJ' 'GET' 'BET' 'REC' 'ESP' 'VAL'
 'BIL' 'ALM' 'AMA' 'MAL' 'XER' 'ZAR' 'TEN' 'LEV' 'HER' 'RSO' 'GRA' 'RAY'
 'CEL' 'ELC' 'EIB' 'COR' 'LAS' 'GRA' 'BEL' 'YB' 'BAS' 'AAR' 'SIO' 'LUZ' 'VAD'
 'XAM' 'GAL' 'THU' 'SER' 'LAU' 'LUG']
```

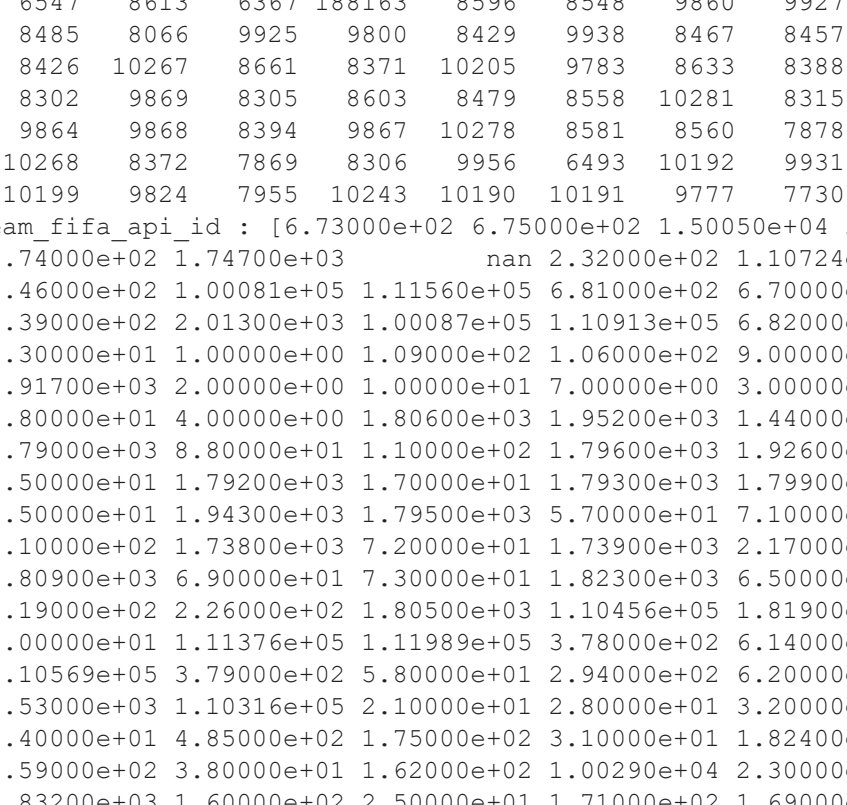
In [113]:

```
for col in df5.columns:
    df5[col].replace({'?':np.nan},inplace=True)
```

In [114]:

```
sns.heatmap(df5.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

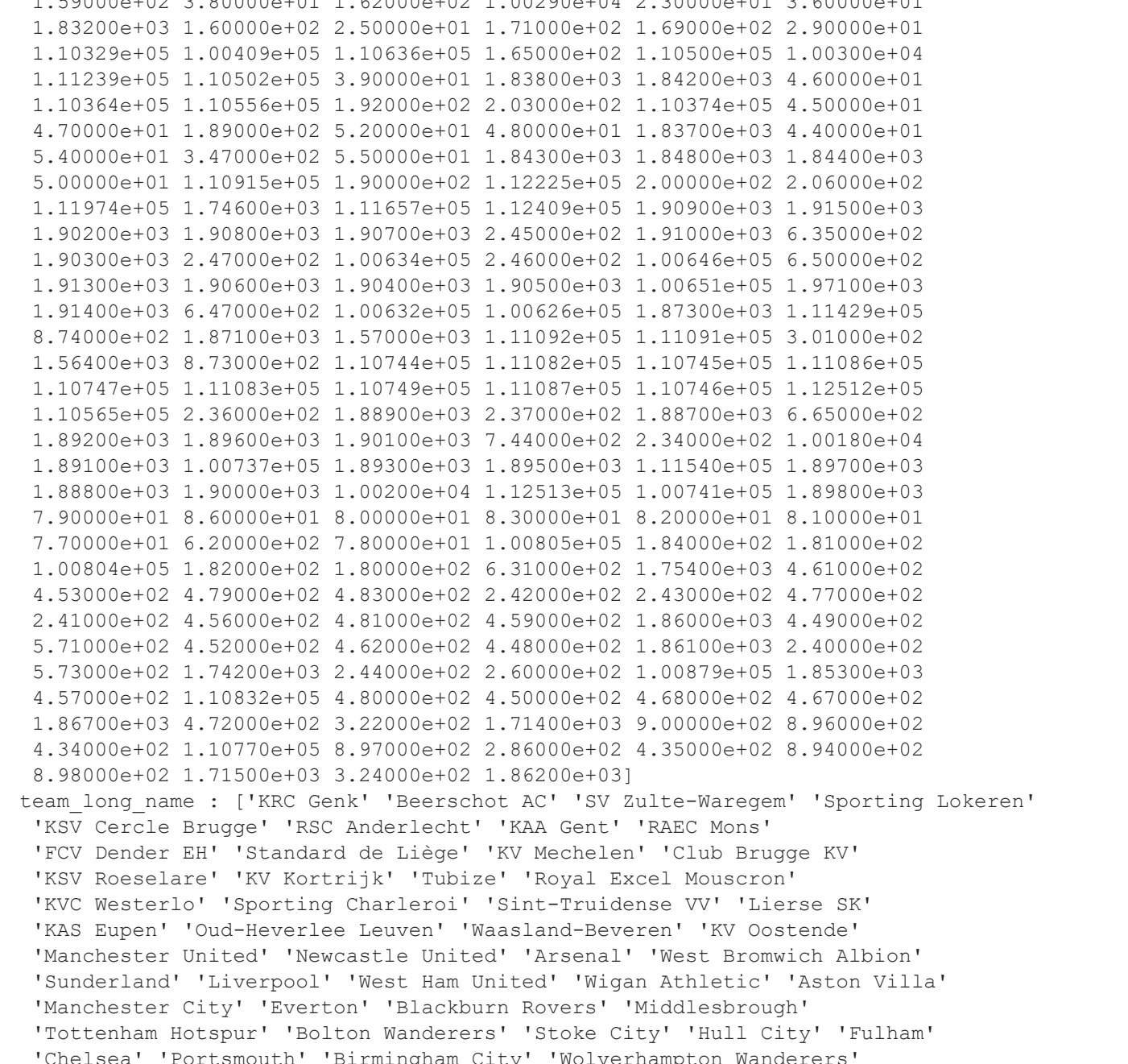
Out[114]: <AxesSubplot:>



In [115]:

```
#Once again, the way to deal with this amount of missing data is tuple deletion to be done later
plt.figure(figsize=(10,10))
sns.heatmap(df5.corr(),cbar=True,annot=True,cmap='Blues')
```

Out[115]: <AxesSubplot:>



In [116]:

```
df6 = pd.read_csv('Team_Attributes.csv')
```

In [117]:

```
df6.head()
```

Out[117]:

	id	team_fifa_api_id	team_api_id	date	buildUpPlaySpeed	buildUpPlaySpeedClass	buildUpPlayDribbling	buildUpPlayDribblingClass	buil
0	1	434	9930	2010-02-22 00:00:00	60	Balanced	NaN	Little	
1	2	434	9930	2014-09-19 00:00:00	52	Balanced	48.0	Normal	
2	3	434	9930	2015-09-10 00:00:00	47	Balanced	41.0	Normal	
3	4	77	8485	2010-02-22 00:00:00	70	Fast	NaN	Little	
4	5	77	8485	2011-02-22 00:00:00	47	Balanced	NaN	Little	

5 rows × 25 columns

In [118]:

```
df6.describe()
```

Out[118]:

	id	team_fifa_api_id	team_api_id	buildUpPlaySpeed	buildUpPlayDribbling	buildUpPlayPassing	chanceCreationPassing	chanc
count	1458.000000	1458.000000	1458.000000	1458.000000	489.000000	1458.000000	1458.000000	
mean	729.500000	17706.982167	9995.727023	52.462277	48.607362	48.490398	52.165295	
std	421.032659	39179.857739	13264.869900	11.545869	9.678290	10.896101	10.360793	
min	1.000000	1.000000	1601.000000	20.000000	24.000000	20.000000	21.000000	
25%	365.250000	110.000000	8457.750000	45.000000	42.000000	40.000000	46.000000	
50%	729.500000	485.000000	8674.000000	52.000000	49.000000	50.000000	52.000000	
75%	1093.750000	1900.000000	9904.000000	62.000000	55.000000	55.000000	59.000000	
max	1458.000000	112513.000000	274581.000000	80.000000	77.000000	80.000000	80.000000	

In [119]:

```
df6.shape
```

Out[119]: (1458, 25)

In [120]:

```
df6.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1458 entries, 0 to 1457
Data columns (total 25 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   id                              1458 non-null   int64  
 1   team_fifa_api_id                1458 non-null   int64  
 2   team_api_id                     1458 non-null   int64  
 3   date                            1458 non-null   object 
 4   buildUpPlaySpeed                1458 non-null   int64  
 5   buildUpPlaySpeedClass           1458 non-null   object 
 6   buildUpPlayDribbling            489 non-null    float64
 7   buildUpPlayDribblingClass       1458 non-null   object 
 8   buildUpPlayPassing              1458 non-null   int64  
 9   buildUpPlayPassingClass         1458 non-null   object 
 10  buildUpPlayPositioningClass     1458 non-null   object 
 11  chanceCreationPassing           1458 non-null   int64  
 12  chanceCreationPassingClass      1458 non-null   object 
 13  chanceCreationCrossing          1458 non-null   int64  
 14  chanceCreationCrossingClass     1458 non-null   object 
 15  chanceCreationShooting          1458 non-null   int64  
 16  chanceCreationShootingClass     1458 non-null   object 
 17  chanceCreationPositioningClass  1458 non-null   object 
 18  defencePressure                 1458 non-null   int64  
 19  defencePressureClass            1458 non-null   object 
 20  defenceAggression               1458 non-null   int64  
 21  defenceAggressionClass          1458 non-null   object 
 22  defenceTeamWidth                1458 non-null   int64  
 23  defenceTeamWidthClass           1458 non-null   object 
 24  defenceDefenderLineClass        1458 non-null   object 
dtypes: float64(1), int64(11), object(13)
memory usage: 284.9+ KB
```

In [121]:

```
#Checking for missing null values
df6.isnull().sum()
```

Out[121]:

```
id                                  0
team_fifa_api_id                    0
team_api_id                         0
date                                0
buildUpPlaySpeed                    0
buildUpPlaySpeedClass               0
buildUpPlayDribbling              969
buildUpPlayDribblingClass           0
buildUpPlayPassing                  0
buildUpPlayPassingClass             0
buildUpPlayPositioningClass         0
chanceCreationPassing               0
chanceCreationPassingClass          0
chanceCreationCrossing              0
chanceCreationCrossingClass         0
chanceCreationShooting              0
chanceCreationShootingClass         0
chanceCreationPositioningClass      0
defencePressure                     0
defencePressureClass                0
defenceAggression                   0
defenceAggressionClass              0
defenceTeamWidth                    0
defenceTeamWidthClass               0
defenceDefenderLineClass            0
dtype: int64
```

In [122]:

```
#Checking for wrong entries like symbols -,?,#,*,etc.
for col in df6.columns:
    print('{} : {}'.format(col,df6[col].unique()))
```

```
id : [   1    2    3 ... 1456 1457 1458]
team_fifa_api_id : [   434     77    614     47   1901    650    245   1861    229   1530
 111082 111989    159 112513      1      2     39    448    240 100409
     57   1906    241   1848    896     32     21    675   1897   1889
   1714    234     88      3   1926   1898    160    189      4     59
     22     23 111376   1943   1896    190    378   1796    210   1842
    647   1961 112409 110364    450     78   1750 110915      5    192
    231   1867 110747   1799 110502    635    242 110569    180    181
    182    467 110500   1824    468   1746    162    452  10020   2013
      7 111271   1971     79     31 100626   1915    236   1903    286
    894    246 110374 110636     25 111657    144    673 110556    674
   1860 110316   1888 110744 100632 110565 111429 110832    322   1805
    165     62     28    184    485     80 100634 100879    166     81
  10029   1952 111239     44    620 110745     45     29   1832     82
 111083 100081 110724    472   1862     46   1738   1739    347    873
 111091   1871     95  10018     64   1853    239     65      9   1844
    301   2007    217    897     66    169    573    453     10     11
   1893    219     68     12     69   1747     70   1900     83 111560
   1904   1891   1823     71 100737   1910     13     72   1792 112225
    171    477   1564 111540    682    479 100087   1892  10030   1843
     73     50   1754   1914 100741    200 111086 111087 110746   1570
   1790    247     15    456     86    480   1793    449    243    457
    571    203    379     74    744   1905   1838   1902    546     52
    631    874   1819   1837 111974     58   1913     34    324    481
 110770 111092    226     17 100646    670    237    459     48    898
 100805 110329 100804    680    232   1806    106   1960    260   1715
     54     18   1809    294   1908     55   1895    461 110456    462
    206     36    483   1909   1887    665 100651 110913   1795     38
    109     19    681   1917   1907   1873    175    110    435   1742
    900 110749    244 112512  15005]
team_api_id : [  9930   8485   8576   8564  10215  10217   8593   9865   8635   8121
   8322 108893   9912 158085   9825  10252   8524   8315   9906   8406
   8583  10229   8634   9976   9931   8178   9823   9993  10211   9807
   6493   9772   8658   8655   8483   8613   9911   9857   8559   9827
   9789   9788   4170   8678  10264   9858   8521   8191   7819   8529
   7788   8344 208931   8530   9910   9925   9984   9880   8455   8533
   8342   7869   2186   9826   8262   8526   9783   9836   8284   9938
   8457   8372   9776   9810  10268   8534   8398   8558   7842   6351
   8668   4087  10218   8596   8722   6631   8674   9773   9908   9824
  10243  10235   8535   8194   8358   9891   9879   9987  10233   9991
   8305   6391   9764   8569   6433   8019   8020   7878   9956   9855
   8357   9747   9790   8429   9904   9860   9791  10278   8177  10251
   8226   8667   8234   8636   8066   1957   9885   8350   8295   8597
   8245   8571   8203   8306   7730   8543   9746   8682   9888   2182
   8030   8673   8197   6421   8588   8581   9989   8639   8650   8537
   8244   9994   8689  10199   9748   9905   9864   8661   8456  10260
  10212   8592   8550   8549   9829   9998  10249   8348   9927 274581
   9996   9761  10214   8481   9830   9809   8464  10261   9831   9850
   6269   8165   8388   8242   2033   8573   8371   1773   6403   8460
   8540   9847  10167   8426   6413   6547   9878   8028   8033   8023
   8031   2183   8462   8640  10172   8696   8548   8370   9798   8603
   8633   8560   8479   8690   9837   9851   7841  10219   8551   9803
   9999   8686   8649   1601   9853   9882   7943   7794  10228  10189
   9777   8302  10179   8025   9874   8466   8614   9986   9768   9869
   9875  10190   9800   8152   8467   9997   9985  10194   8472  10003
   9867  10191   9804   8586   9941  10242   8611   8600   9771  10267
   9873  10281   9876  10269  10205   8277   7844  10238   9839   8475
   9817   8697   8659   8654  10001   8024   8528   8525  10265   8721
   8602   7955   9868  10192   8021   8394   8027  10000]
date : ['2010-02-22 00:00:00' '2014-09-19 00:00:00' '2015-09-10 00:00:00'
 '2011-02-22 00:00:00' '2012-02-22 00:00:00' '2013-09-20 00:00:00']
buildUpPlaySpeed : [60 52 47 70 58 62 59 65 45 48 30 53 38 56 40 31 35 55 42 46 50 23 41 39
 69 66 75 25 67 63 64 57 68 43 24 36 61 73 37 51 44 49 71 74 76 54 32 80
 34 72 29 78 33 26 28 20 77]
buildUpPlaySpeedClass : ['Balanced' 'Fast' 'Slow']
buildUpPlayDribbling : [nan 48. 41. 64. 57. 70. 53. 47. 40. 43. 46. 61. 49. 66. 51. 32. 37. 45.
 52. 50. 38. 55. 35. 63. 30. 29. 34. 24. 39. 31. 60. 44. 36. 56. 54. 33.
 59. 58. 42. 69. 62. 67. 65. 77. 28. 68. 71. 26. 27. 74.]
buildUpPlayDribblingClass : ['Little' 'Normal' 'Lots']
buildUpPlayPassing : [50 56 54 70 52 62 45 53 40 30 44 35 47 55 66 38 33 39 65 41 51 37 48 58
 32 29 26 59 72 36 69 49 31 46 34 25 57 22 28 60 79 75 64 73 61 43 27 74
 67 42 68 77 71 63 24 20 23 80]
buildUpPlayPassingClass : ['Mixed' 'Long' 'Short']
buildUpPlayPositioningClass : ['Organised' 'Free Form']
chanceCreationPassing : [60 54 70 53 45 40 56 51 65 48 55 50 66 30 67 39 58 57 68 35 52 33 49 41
 28 44 63 38 36 61 47 59 37 77 34 21 43 69 32 42 72 46 62 71 64 73 31 80
 76 29]
chanceCreationPassingClass : ['Normal' 'Risky' 'Safe']
chanceCreationCrossing : [65 63 70 48 50 68 72 35 34 38 45 60 20 53 36 57 51 67 55 52 64 54 62 40
 33 59 44 30 47 49 78 73 69 74 56 25 24 31 66 46 37 71 61 58 43 26 42 77
 39 41 76 27 80 23 75 32]
chanceCreationCrossingClass : ['Normal' 'Lots' 'Little']
chanceCreationShooting : [55 64 70 52 57 63 50 75 69 60 65 66 44 38 67 46 39 30 40 47 48 35 43 42
 56 34 51 59 37 36 79 54 49 72 53 68 61 22 41 58 45 76 73 62 80 78 33 32
 71 28 31 23 77 24 27 74 29]
chanceCreationShootingClass : ['Normal' 'Lots' 'Little']
chanceCreationPositioningClass : ['Organised' 'Free Form']
defencePressure : [50 47 60 40 42 41 49 30 45 38 48 58 52 39 46 57 65 68 64 70 37 53 44 25
 36 35 51 43 55 54 59 33 34 67 66 61 72 32 28 27 23 24 63 31 26 29 56 62]
defencePressureClass : ['Medium' 'Deep' 'High']
defenceAggression : [55 44 70 47 40 42 45 35 50 49 57 30 38 62 58 65 53 68 43 48 32 54 37 39
 41 66 63 67 60 34 61 51 46 72 52 59 56 33 69 71 64 27 28 36 24 31 29]
defenceAggressionClass : ['Press' 'Double' 'Contain']
defenceTeamWidth : [45 54 70 52 60 63 30 50 53 49 65 61 62 64 59 67 51 35 36 58 57 48 37 55
 68 42 56 41 40 66 47 38 46 43 39 44 33 69 32 34 29 31 73]
defenceTeamWidthClass : ['Normal' 'Wide' 'Narrow']
defenceDefenderLineClass : ['Cover' 'Offside Trap']
```

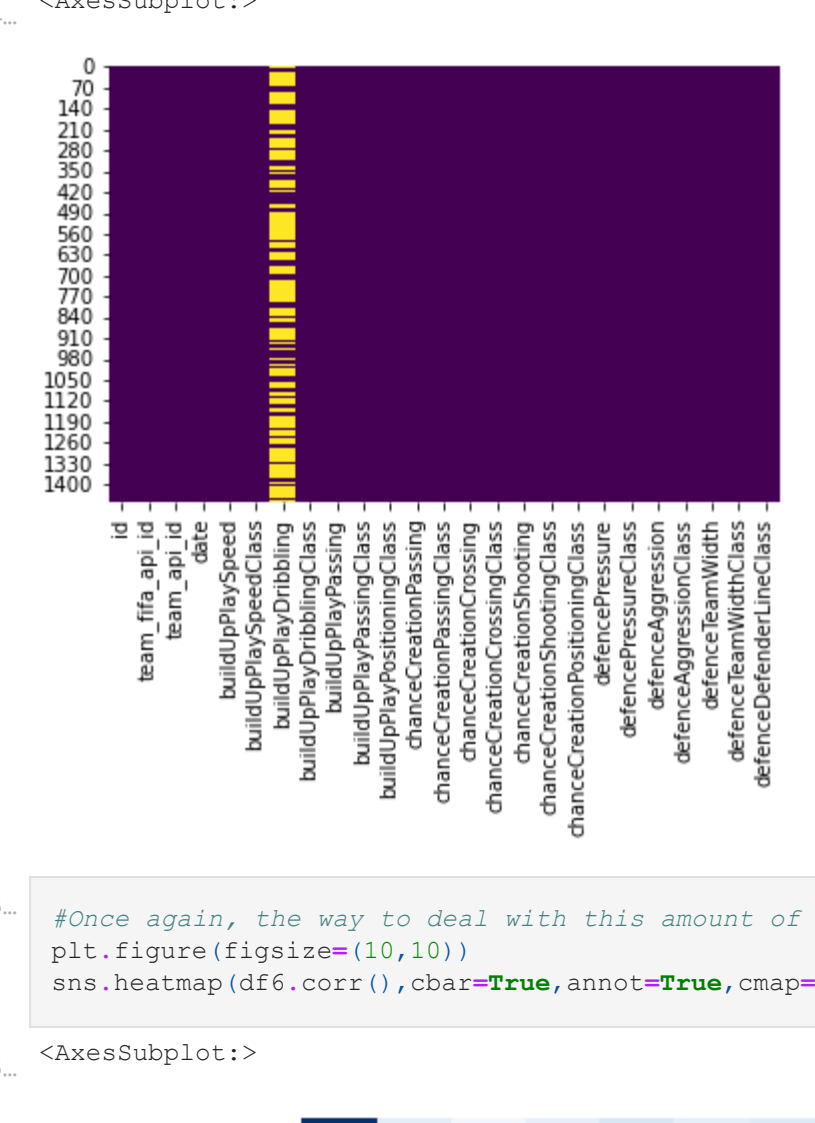
In [123…

```
for col in df6.columns:
    df6[col].replace({'?':np.nan},inplace=True)
```

In [124…

```
sns.heatmap(df6.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

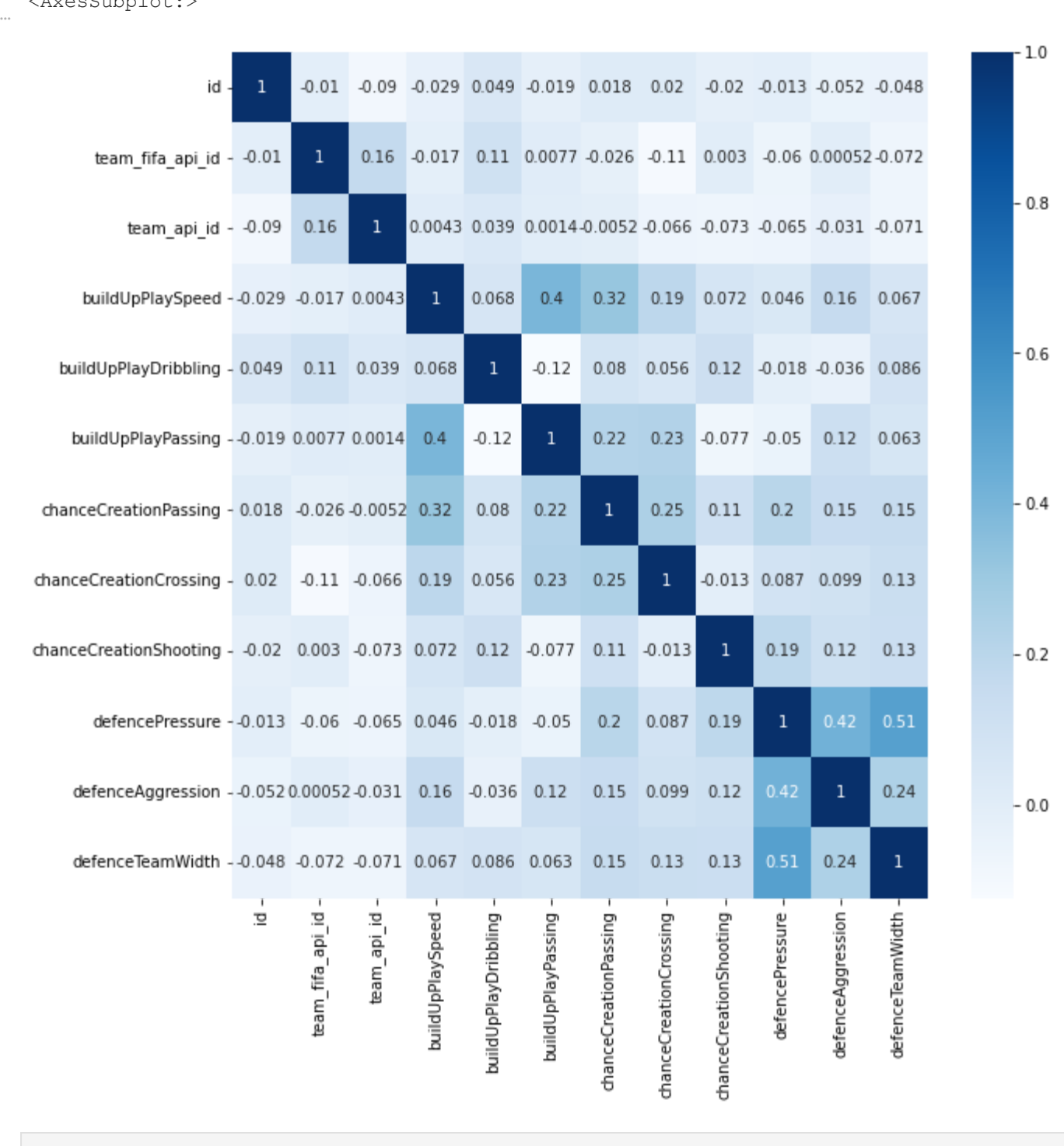
Out[124… <AxesSubplot:>



In [125…

```
#Once again, the way to deal with this amount of missing data is tuple deletion to be done later
plt.figure(figsize=(10,10))
sns.heatmap(df6.corr(),cbar=True,annot=True,cmap='Blues')
```

Out[125… <AxesSubplot:>



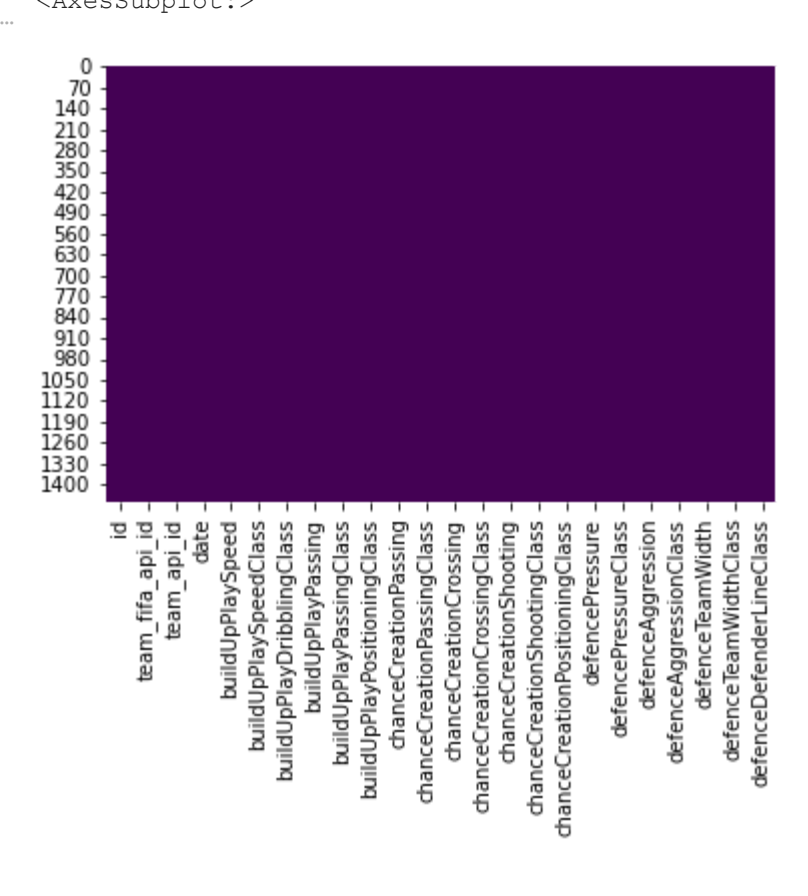
In [130…

```
#Since there is one column with a large amount of missing data, we deem it to be insignificant and delete that
df6.drop(['buildUpPlayDribbling'],axis=1,inplace=True)
```

In [131…

```
sns.heatmap(df6.isnull(),cbar=False,cmap='viridis')
#Used to get the heatmap for missing or incorrect data
```

Out[131… <AxesSubplot:>



This concludes EDA for all tables involved in dataset