

Game of Thrones Analysis

```
In [1]: # Loading required libraries
import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
```

```
In [2]: got=pd.read_csv("C:\\Users\\Pranav\\Desktop\\DATA SCIENCE DATA\\CVC file\\Game of Thrones\\battles.csv")
got.head()
```

Out[2]:

	name	year	battle_number	attacker_king	defender_king	attacker_1	attacker_2	attacker_3	attacker_4	defender_1	...	major_death	major_capture	attacker_size	defender_size	attacker_commander	defender_commander
0	Battle of the Golden Tooth	298	1	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Tully ...	1.0	0.0	15000.0	4000.0	Jaime Lannister	Clement Piper, Va
1	Battle at the Mummer's Ford	298	2	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Baratheon ...	1.0	0.0	NaN	120.0	Gregor Clegane	Beric Dondar
2	Battle of Riverrun	298	3	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Tully ...	0.0	1.0	15000.0	10000.0	Jaime Lannister, Andros Brax	Edmure Tully, Ty Blackw
3	Battle of the Green Fork	298	4	Robb Stark	Joffrey/Tommen Baratheon		Stark	NaN	NaN	NaN	Lannister ...	1.0	1.0	18000.0	20000.0	Roose Bolton, Wylis Manderly, Medger Cerwyn, H...	Tywin Lannist Gregor Cleg Kevan Lanni
4	Battle of the Whispering Wood	298	5	Robb Stark	Joffrey/Tommen Baratheon		Stark	Tully	NaN	NaN	Lannister ...	1.0	1.0	1875.0	6000.0	Robb Stark, Brynden Tully	Jaime Lanni

5 rows × 25 columns

```
In [3]: # shape of dataset
got.shape
```

Out[3]: (38, 25)

```
In [4]: # information about dataset
got.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38 entries, 0 to 37
Data columns (total 25 columns):
# Column Non-Null Count Dtype
---  ---
0 name 38 non-null object
1 year 38 non-null int64
2 battle_number 38 non-null int64
3 attacker_king 36 non-null object
4 defender_king 36 non-null object
5 attacker_1 37 non-null object
6 attacker_2 19 non-null object
7 attacker_3 3 non-null object
8 attacker_4 2 non-null object
9 defender_1 37 non-null object
10 defender_2 2 non-null object
11 defender_3 0 non-null float64
12 defender_4 0 non-null float64
13 attacker_outcome 37 non-null object
14 battle_type 37 non-null object
15 major_death 37 non-null float64
16 major_capture 37 non-null float64
17 attacker_size 24 non-null float64
18 defender_size 19 non-null float64
19 attacker_commander 37 non-null object
20 defender_commander 28 non-null object
21 summer 37 non-null float64
22 location 37 non-null object
23 region 38 non-null object
24 note 5 non-null object
dtypes: float64(7), int64(2), object(16)
memory usage: 7.5+ KB
```

```
In [5]: #Describe of dataset
got.describe()
```

Out[5]:

	year	battle_number	defender_3	defender_4	major_death	major_capture	attacker_size	defender_size	summer
count	38.000000	38.000000	0.0	0.0	37.000000	37.000000	24.000000	19.000000	37.000000
mean	299.105263	19.500000	NaN	NaN	0.351351	0.297297	9942.541667	6428.157695	0.702703
std	0.689280	11.113055	NaN	NaN	0.483978	0.463373	20283.092065	6225.182106	0.463373
min	298.000000	1.000000	NaN	NaN	0.000000	0.000000	20.000000	100.000000	0.000000
25%	299.000000	10.250000	NaN	NaN	0.000000	0.000000	1375.000000	1070.000000	0.000000
50%	299.000000	19.500000	NaN	NaN	0.000000	0.000000	4000.000000	6000.000000	1.000000
75%	300.000000	28.750000	NaN	NaN	1.000000	1.000000	8250.000000	10000.000000	1.000000
max	300.000000	38.000000	NaN	NaN	1.000000	1.000000	100000.000000	20000.000000	1.000000

```
In [6]: # find out null value in dataset
got.isnull().sum()
```

Out[6]:

name	0
year	0
battle_number	0
attacker_king	2
defender_king	3
attacker_1	0
attacker_2	28
attacker_3	35
attacker_4	36
defender_1	1
defender_2	36
defender_3	38
defender_4	38
attacker_outcome	1
battle_type	1
major_death	1
major_capture	1
attacker_size	14
defender_size	19
attacker_commander	1
defender_commander	10
summer	1
location	1
region	0
note	33
dtype: int64	

```
In [9]: # Replacing the columns
got.rename(columns={'attacker_1':'Primary_Attacker'},inplace=True)
got.head()
```

Out[9]:

	name	year	battle_number	attacker_king	defender_king	Primary_Attack	attacker_2	attacker_3	attacker_4	defender_1	...	major_death	major_capture	attacker_size	defender_size	attacker_commander	defender_commander
0	Battle of the Golden Tooth	298	1	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Tully ...	1.0	0.0	15000.0	4000.0	Jaime Lannister	Clement Piper
1	Battle at the Mummer's Ford	298	2	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Baratheon ...	1.0	0.0	NaN	120.0	Gregor Clegane	Beric Dondarrion
2	Battle of Riverrun	298	3	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Tully ...	0.0	1.0	15000.0	10000.0	Jaime Lannister, Andros Brax	Edmure Tully, Ty Blackwater
3	Battle of the Green Fork	298	4	Robb Stark	Joffrey/Tommen Baratheon		Stark	NaN	NaN	NaN	Lannister ...	1.0	1.0	18000.0	20000.0	Roose Bolton, Wylis Manderly, Medger Cerwyn, H...	Tywin Lannister Gregor Clegane Kevan Lannister
4	Battle of the Whispering Wood	298	5	Robb Stark	Joffrey/Tommen Baratheon		Stark	Tully	NaN	NaN	Lannister ...	1.0	1.0	1875.0	6000.0	Robb Stark, Brynden Tully	Jaime Lannister

5 rows × 25 columns

```
In [10]: got.rename(columns={'attacker_2':'Primary_Defender'},inplace=True)
got.head()
```

Out[10]:

	name	year	battle_number	attacker_king	defender_king	Primary_Attack	Primary_Defender	attacker_3	attacker_4	defender_1	...	major_death	major_capture	attacker_size	defender_size	attacker_commander	defender_commander
0	Battle of the Golden Tooth	298	1	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Tully ...	1.0	0.0	15000.0	4000.0	Jaime Lannister	Clement Piper
1	Battle at the Mummer's Ford	298	2	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Baratheon ...	1.0	0.0	NaN	120.0	Gregor Clegane	Beric Dondarrion
2	Battle of Riverrun	298	3	Joffrey/Tommen Baratheon	Robb Stark	Lannister		NaN	NaN	NaN	Tully ...	0.0	1.0	15000.0	10000.0	Jaime Lannister, Andros Brax	Edmure Tully, Ty Blackwater
3	Battle of the Green Fork	298	4	Robb Stark	Joffrey/Tommen Baratheon		Stark	NaN	NaN	NaN	Lannister ...	1.0	1.0	18000.0	20000.0	Roose Bolton, Wylis Manderly, Medger Cerwyn, H...	Tywin Lannister Gregor Clegane Kevan Lannister
4	Battle of the Whispering Wood	298	5	Robb Stark	Joffrey/Tommen Baratheon		Stark	Tully	NaN	NaN	Lannister ...	1.0	1.0	1875.0	6000.0	Robb Stark, Brynden Tully	Jaime Lannister

5 rows × 25 columns

```
In [11]: # value counts in columns
got['attacker_king'].value_counts()
```

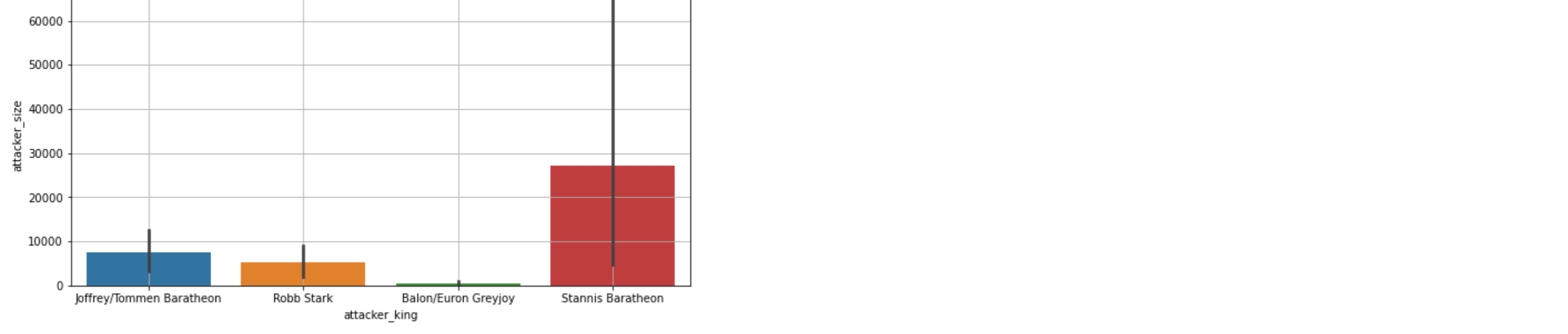
Out[11]: Joffrey/Tommen Baratheon 14  
Robb Stark 10  
Balon/Euron Greyjoy 7  
Stannis Baratheon 5  
Name: attacker\_king, dtype: int64

```
In [12]: got['location'].value_counts()
```

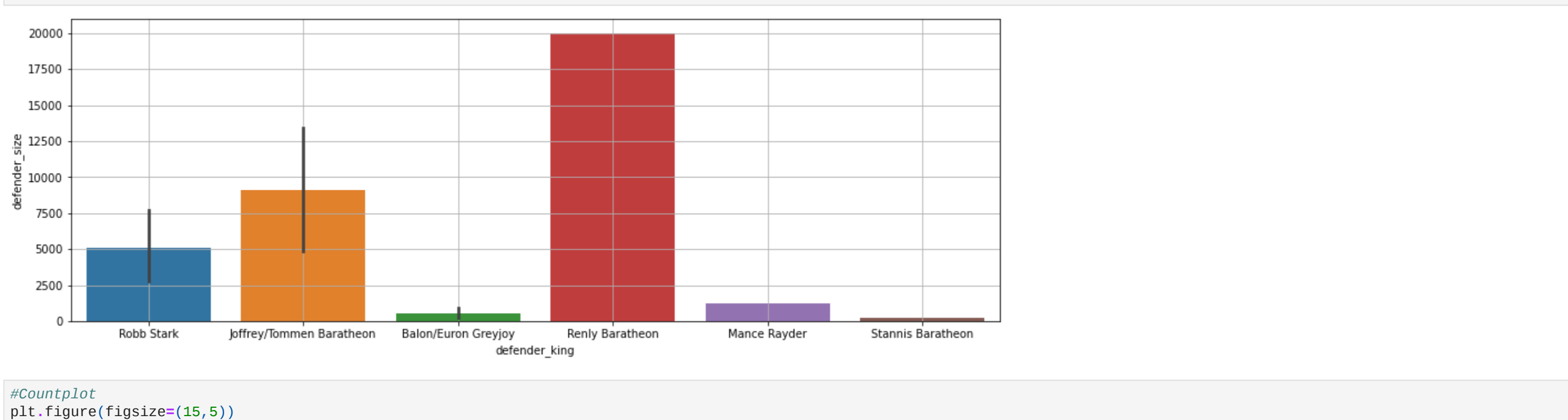
Out[12]:

Riverrun	3
Winterfell	3
Storm's End	2
Harrenhal	2
Darry	2
Moat Cailin	2
Deepwood Motte	2
Torrhen's Square	2
Golden Tooth	1
Seagard	1
Castle Black	1
Shield Islands	1
Saltpans	1
Ruby Ford	1
Ryansport, Vinetown, Starfish Harbor	1
Dragonstone	1
The Twins	1
Red Fork	1
Duskendale	1
King's Landing	1
Crag	1
Mummer's Ford	1
Oxcross	1
Stony Shore	1
Whispering Wood	1
Green Fork	1
Ravenree	1
Name: location, dtype: int64	

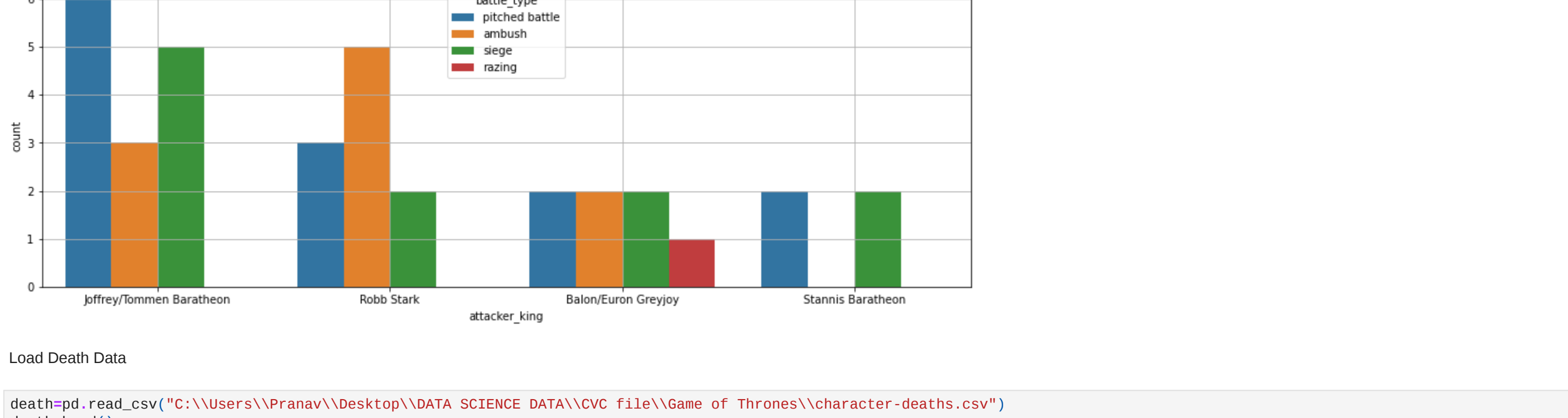
```
In [14]: # Barplot
plt.figure(figsize=(10,5))
sns.barplot(x='attacker_king',y='attacker_size',data=got)
plt.grid(True)
plt.show()
```



```
In [16]: plt.figure(figsize=(15,5))
sns.barplot(x='defender_king',y='defender_size',data=got)
plt.grid(True)
plt.show()
```



```
In [20]: #Countplot
plt.figure(figsize=(15,5))
sns.countplot(x=got['attacker_king'],hue=got['battle_type'])
plt.grid(True)
plt.show()
```



Load Death Data

```
In [22]: death=pd.read_csv("C:\\Users\\Pranav\\Desktop\\DATA SCIENCE DATA\\CVC file\\Game of Thrones\\character-deaths.csv")
death.head()
```

Out[22]:

	Name	Alliances	Death Year	Book of Death	Death Chapter	Book Intro Chapter	Gender	Nobility	GoT	CoK	SoS	FiC	DwD
0	Addam Marbrand	Lannister		NaN	NaN	49.0	1	1	1	1	1	1	0
1	Aegon Frey (Jinglebell)	None	299.0	3.0	51.0	49.0	1	1	0	0	1	0	0
2	Aegon Targaryen	House Targaryen		NaN	NaN	5.0	1	1	0	0	0	0	1
3	Adrack Humble	House Greyjoy	300.0	5.0	20.0	20.0	1	1	0	0	0	0	1
4	Aemon Costayne	Lannister		NaN	NaN	NaN	1	1	0	0	1	0	0

```
In [23]: # shape of dataset
death.shape
```

Out[23]: (917, 13)

```
In [24]: # information about dataset
death.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 917 entries, 0 to 916
Data columns (total 13 columns):
# Column Non-Null Count Dtype
---  ---
0 Name 917 non-null object
1 Alliances 917 non-null object
2 Death Year 389 non-null float64
3 Book of Death 307 non-null float64
4 Death Chapter 299 non-null float64
5 Book Intro Chapter 905 non-null float64
6 Gender 917 non-null int64
7 Nobility 917 non-null int64
8 GoT 917 non-null int64
9 SoS 917 non-null int64
10 FiC 917 non-null int64
11 DwD 917 non-null int64
dtypes: float64(4), int64(7), object(2)
memory usage: 93.3+ KB
```

```
In [25]: #Describe of dataset
death.describe()
```

Out[25]:

	Death Year	Book of Death	Death Chapter	Book Intro Chapter	Gender	Nobility	GoT	CoK	SoS	FiC	DwD
count	305.000000	307.000000	299.000000	905.000000	917.000000	917.000000	917.000000	917.000000	917.000000	917.000000	917.000000
mean	299.157377	2.928339	40.070234	28.861878	0.828790	0.468920	0.272628	0.353326	0.424209	0.272628	0.284624
std	0.703483	1.326482	20.470270	20.165788	0.376898	0.499305	0.445554	0.478264	0.494492	0.445554	0.451481
min	297.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	299.000000	2.000000	25.500000	11.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	299.000000	3.000000	39.000000	27.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
75%	300.000000	4.000000	57.000000	43.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
max	300.000000	5.000000	80.000000	80.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

```
In [26]: # find out null value in dataset
death.isnull().sum()
```

Out[26]:

Name	0
Alliances	0
Death Year	612
Book of Death	618
Death Chapter	618
Book Intro Chapter	12
Gender	0
Nobility	0
GoT	0
CoK	0
SoS	0
FiC	0
DwD	0
dtype: int64	

```
In [27]: #value counts in columns
death['Gender'].value_counts()
```

Out[27]: 1 760  
0 157  
Name: Gender, dtype: int64

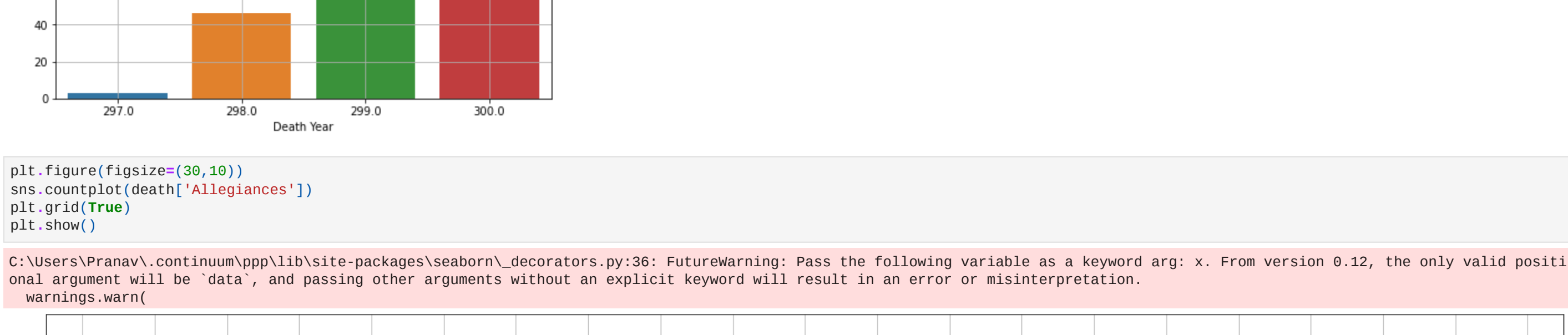
```
In [28]: death['Nobility'].value_counts()
```

Out[28]: 0 487  
1 438  
Name: Nobility, dtype: int64

```
In [30]: # Countplot
plt.figure(figsize=(8,5))
sns.countplot(death['Death Year'])
plt.grid(True)
plt.show()
```

C:\Users\Pranav\continuum\ppp\lib\site-packages\seaborn\decorators.py:36: FutureWarning: Pass the following variable 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()



```
In [36]: plt.figure(figsize=(30,10))
sns.countplot(death['Alliances'])
plt.grid(True)
plt.show()
```

C:\Users\Pranav\continuum\ppp\lib\site-packages\seaborn\decorators.py:36: FutureWarning: Pass the following variable 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()

