Game of Thrones

Table of contents

```
season = "season_4"
```

```
# Injected Parameters
season = "season_3"
```

```
import pandas as pd
from pathlib import Path
import os
from IPython.display import Markdown, display

# List of seasons to process
seasons = ["season_1", "season_2", "season_3", "season_4",
           "season_5", "season_6", "season_7", "season_8"]

# Construct the base data directory path
data_dir = Path("../Data/")

# Loop through each season
file_path = os.path.join(data_dir, season + '.csv')
print(file_path)
df = pd.read_csv(file_path)
```

```
..\Data\season_3.csv
```

```
display(Markdown(f"""
### Game of Thrones - {season} summary in numbers
"""))
```

0.1 Game of Thrones - season_3 summary in numbers

0.2 (*Warning:* spoilers ahead)

0.3 Overview

(From the Wikipedia) Game of Thrones is an American fantasy drama television series created by David Benioff and D. B. Weiss for HBO. It is an adaptation of A Song of Ice and Fire, a series of fantasy novels by George R. R. Martin, the first of which is A Game of Thrones.

Set on the fictional continents of Westeros and Essos, Game of Thrones has a large ensemble cast and follows several story arcs throughout the course of the show. A major arc concerns the Iron Throne of the Seven Kingdoms of Westeros through a web of political conflicts among the noble families either vying to claim the throne or fighting for independence from it. Another focuses on the last descendant of the realm's deposed ruling dynasty, who has been exiled to Essos and is plotting a return to the throne. A third story arc follows the Night's Watch, a military order defending the realm against threats from the North.

```
display(Markdown(f"""
### {season} - episode descriptions
"""))
```

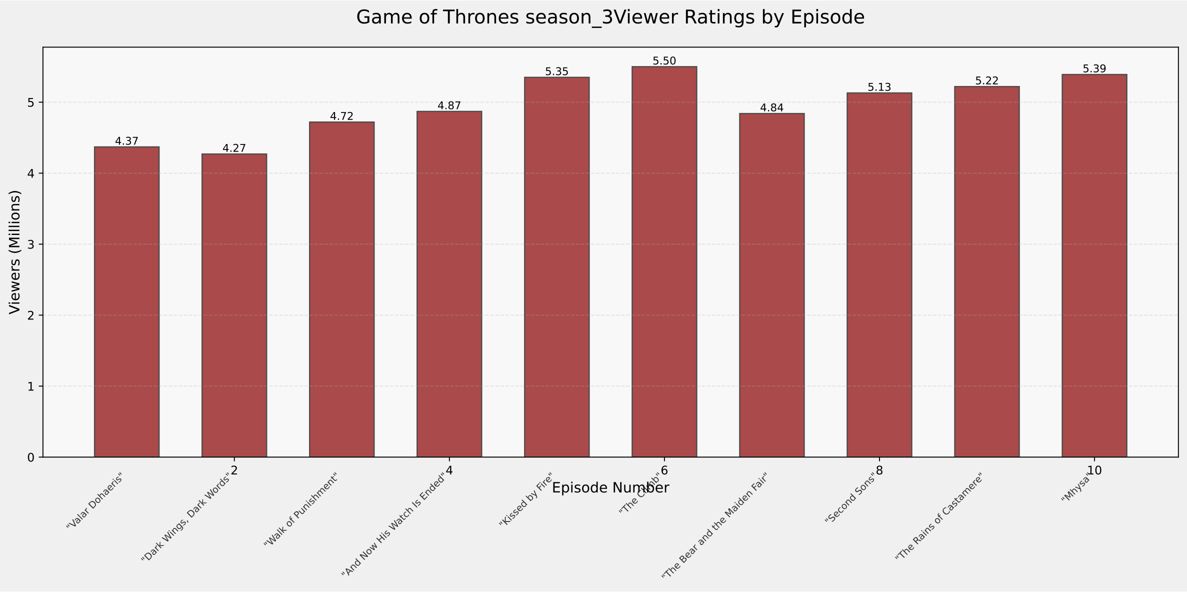
0.4 season_3 - episode descriptions

```
for description in df["description"]:
    display(Markdown(f"""
    > {description}

    """))  # Added extra newline for better spacing
```

```
> After the White Walkers' attack, the surviving Night's Watchmen, including Samwell Tarly ar

> Brienne continues escorting Jaime Lannister across the Riverlands to King's Landing. He eng

> Robb and Catelyn attend Hoster Tully's funeral in Riverrun. Tywin Lannister sends Littlefi

> Theon's "rescuer" betrays him, returning him to the prison for more torture. After a faile

> Tyrion, knowing the Crown is deeply in debt to the Iron Bank, persuades Olenna to pay for

> In King's Landing, Tywin coerces Olenna into an agreement to marry Cersei to Loras. Cersei

> In the North, Jojen tells Bran they must not stop at Castle Black, but travel beyond the Wa

> King's Landing hosts Tyrion and Sansa's wedding. Cersei shows her displeasure toward Marga

> Sam, Gilly, and her son arrive at the Wall. While hiding in a tower during a thunderstorm,

> The massacre's mastermind is revealed to be Tywin Lannister, who rewards House Frey with t
```

You can see how the viewership of the episodes changed in Figure 1.

```python
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.ticker import MaxNLocator

# Create the plot
plt.figure(figsize=(14, 7))

# Create histogram-style bars
bars = plt.bar(df['no_season'], df['viewers'],
               color='darkred',
               alpha=0.7,
               edgecolor='black',
               width=0.6)

# Customize the plot
plt.title('Game of Thrones ' + str(season) + 'Viewer Ratings by Episode', fontsize=16, pad=2
plt.xlabel('Episode Number', fontsize=12)
plt.ylabel('Viewers (Millions)', fontsize=12)
plt.grid(True, linestyle='--', alpha=0.3, axis='y')

# Ensure x-axis shows whole numbers for episode numbers
plt.gca().xaxis.set_major_locator(MaxNLocator(integer=True))

# Add value labels on top of each bar
for bar in bars:
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width()/2., height,
             f'{height:.2f}',
             ha='center', va='bottom',
             fontsize=9)

# Add episode titles below each bar
for i, row in df.iterrows():
    plt.text(row['no_season'], -0.2,
             row['title'],
             rotation=45,
             ha='right',
             va='top',
             fontsize=8,
             alpha=0.8)

# Adjust margins to accommodate episode titles
plt.subplots_adjust(bottom=0.3)

# Add some thematic styling
plt.gca().set_facecolor('#f8f8f8')
plt.gcf().set_facecolor('#f0f0f0')

plt.tight_layout()
plt.show()
```



```
from IPython.display import Markdown
import pandas as pd

# Calculate total viewers and average
total_viewers = df['viewers'].sum()
avg_view = total_viewers / len(df)

# Filter episodes with above-average viewership
above_avg = df[df['viewers'] >= avg_view]

# Create Markdown table
table_header = "| Episode | Title | Viewers (Millions) |\n|---------|-------|------------------
table_rows = []

for _, row in above_avg.iterrows():
    table_rows.append(f"| {int(row['no_season'])} | {row['title']} | {row['viewers']:.2f} |"

# Combine header and rows
markdown_table = f"""
**Episodes with Above-Average Viewership (Average: {avg_view:.2f} million viewers)**

{table_header}
{"\n".join(table_rows)}
```

```
"""

# Display the table
display(Markdown(markdown_table))
```

Episodes with Above-Average Viewership (Average: 4.97 million viewers)

Episode	Title	Viewers (Millions)
5	“Kissed by Fire”	5.35
6	“The Climb”	5.50
8	“Second Sons”	5.13
9	“The Rains of Castamere”	5.22
10	“Mhysa”	5.39