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

# Load titanic dataset
url = r"C:\Users\Prachi\Downloads\titanic_ dataset_final.csv"
df = pd.read_csv(url)
df
```

Out[1]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cal
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	N
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	N
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C1
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	N
...	...	...	...	...	...	...	...	...	...	...	
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	N
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	E
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	N
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C1
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	N

891 rows × 12 columns

In [2]:
```
# Display dataset info
print(df.describe())
```

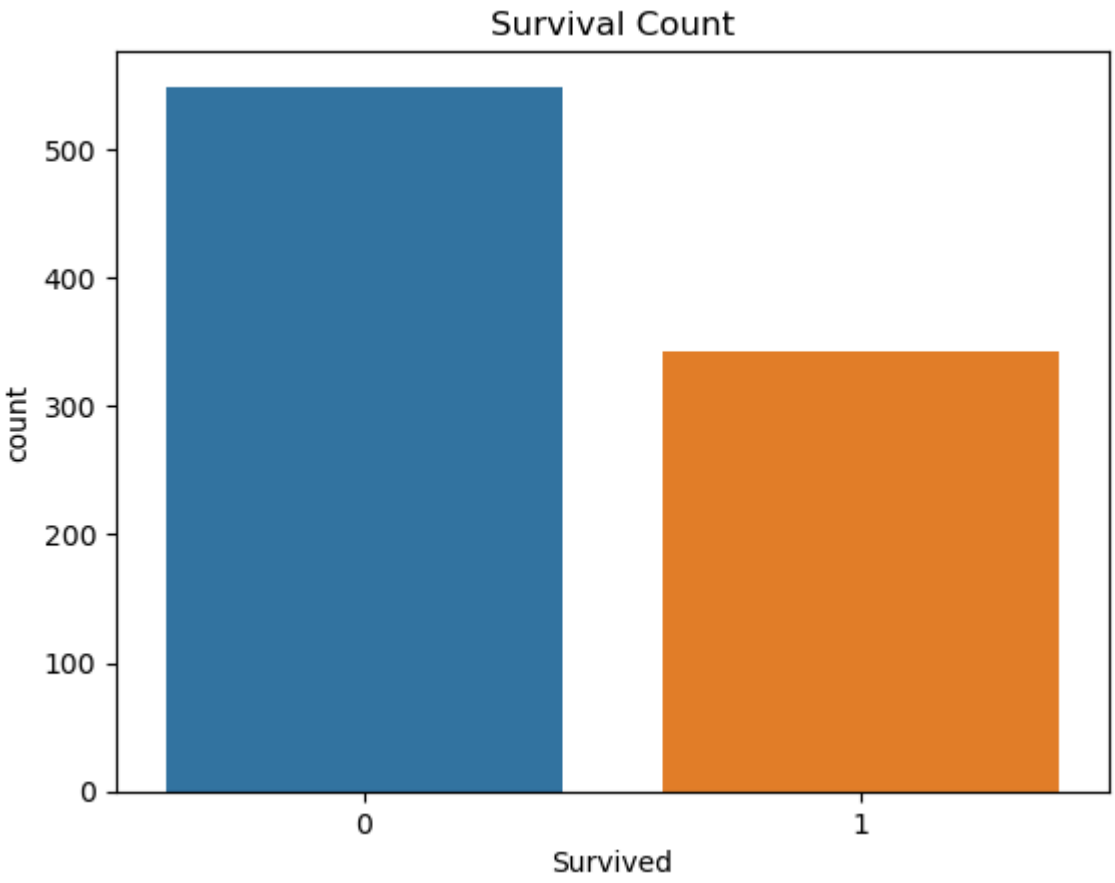
```
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

            Parch        Fare
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
max      6.000000  512.329200
```

In [3]:
```
#Missing Values Check
print("\nMissing Values:\n", df.isnull().sum())
```

```
Missing Values:
 PassengerId      0
Survived          0
Pclass            0
Name              0
Sex               0
Age             177
SibSp             0
Parch             0
Ticket            0
Fare              0
Cabin           687
Embarked          2
dtype: int64
```

In [4]:
```
# Survival Rate Visualization
sns.countplot(x='Survived', data=df)
plt.title("Survival Count")
plt.show()
```



In [6]:

```
import ollama

def generate_insights(df_summary):
    prompt = f"Analyze the dataset summary and provide insights:\n\n{df_summary}"
    response = ollama.chat(model="mistral", messages=[{"role": "user", "content": p
    return response['message']['content']

# Generate AI Insights
summary = df.describe().to_string()
insights = generate_insights(summary)
print("\n 🔹 AI-Generated Insights:\n", insights)
```

🔹 AI-Generated Insights:
This dataset is likely from the Titanic disaster, as it contains variables commo
nly associated with that dataset. Here's a summary of insights based on the provid
ed statistics:

1. The dataset has 891 unique records, which corresponds to the number of passenge
rs on the ship.

2. The mean (average) Survived value is 0.3838, suggesting that slightly less than
half of the passengers survived the disaster.

3. The Pclass variable represents the passenger's ticket class. The mean is 2.308
6, indicating a majority of passengers were traveling in third class (Pclass = 3).

4. The Age column has a mean of 29.69 years and a standard deviation of 14.52 year
s, suggesting a wide range of ages among the passengers.

5. SibSp and Parch represent the number of siblings and parents on board with the
passenger respectively. Both have a mean close to 0, indicating most passengers we
re traveling alone or with very few family members.

6. The Fare column represents the ticket price. The mean fare is 32.20, but it has
a high standard deviation of 49.69, suggesting significant variation in ticket pri
ces.

7. The minimum and maximum values for all variables provide interesting insights:
   - PassengerId: ranges from 1 to 891 (unique passenger IDs)
   - Survived: the minimum value is 0 (indicating death), while the maximum is 1
(indicating survival)
   - Pclass: ranges from 1 to 3, representing the three classes on the Titanic
   - Age: ranges from 0.42 to 80 years old, with a minimum age of an infant and a
maximum age of an elderly passenger
   - SibSp and Parch: range from 0 (no siblings or parents) up to 8 (a passenger t
raveling with a large family)
   - Fare: ranges from 0.00 (likely a mistake) to 512.33, with the lowest fares be
ing for third-class passengers and the highest for first-class passengers.

In [7]:
```
import gradio as gr

def eda_analysis(file):
    df = pd.read_csv(file.name)
    summary = df.describe().to_string()
    insights = generate_insights(summary)
    return insights

# Create Web Interface
demo = gr.Interface(fn=eda_analysis, inputs="file",outputs="text",title="AI-Powered

# Lauch App

demo.launch(share=True)
```

```
* Running on local URL:  http://127.0.0.1:7860
* Running on public URL: https://55aaf602d96a196fb1.gradio.live

This share link expires in 1 week. For free permanent hosting and GPU upgrades, ru
n `gradio deploy` from the terminal in the working directory to deploy to Hugging
Face Spaces (https://huggingface.co/spaces)
```