```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# Load the data
df = pd.read_csv('COVID clinical trials.csv')

# Preprocessing
# Convert Enrollment to numeric
df['Enrollment'] = pd.to_numeric(df['Enrollment'], errors='coerce')

# Define Status Categories
# We consider 'Completed' as a successful outcome.
# 'Terminated' and 'Withdrawn' are unsuccessful outcomes.
# Other statuses like 'Recruiting' are ongoing.
df['Is_Completed'] = (df['Status'] == 'Completed').astype(int)
df['Is_Terminated_Withdrawn'] = df['Status'].isin(['Terminated',
'Withdrawn']).astype(int)

# Grouping 'Funded Bys' into simpler categories
def simplify_funder(funder_str):
    if pd.isna(funder_str):
        return 'Other'
    funders = funder_str.split('|')
    if 'Industry' in funders:
        return 'Industry'
    if 'NIH' in funders:
        return 'NIH'
    if 'U.S. Fed' in funders:
        return 'U.S. Fed'
    return 'Other'

df['Funder_Category'] = df['Funded Bys'].apply(simplify_funder)

# 1. Completion Rate by Phase
# Exclude 'Missing' or irrelevant phases for a cleaner look
valid_phases = ['Phase 1', 'Phase 2', 'Phase 3', 'Phase 4', 'Phase 1|Phase 2',
'Phase 2|Phase 3']
phase_completion = df[df['Phases'].isin(valid_phases)].groupby('Phases')
['Is_Completed'].mean().sort_values(ascending=False)

# 2. Completion Rate by Funder Type
funder_completion = df.groupby('Funder_Category')
```

```
['Is_Completed'].mean().sort_values(ascending=False)

# 3. Enrollment Analysis
# Compare enrollment distribution for Completed vs Terminated/Withdrawn
# Use median to avoid outlier distortion
enrollment_stats = df.groupby('Status')['Enrollment'].agg(['median', 'mean', 'count'])
relevant_status = ['Completed', 'Terminated', 'Withdrawn']
enrollment_comparison = enrollment_stats.loc[relevant_status]

# 4. Terminated/Withdrawn Patterns
# Let's look at the reasons if any (sometimes in other columns or titles, but usually not in this CSV)
# We can look at Phase and Funder distribution for Terminated/Withdrawn trials
terminated_withdrawn_df = df[df['Is_Terminated_Withdrawn'] == 1]
tw_phase_dist = terminated_withdrawn_df['Phases'].value_counts(normalize=True)
tw_funder_dist = terminated_withdrawn_df['Funder_Category'].value_counts(normalize=True)

# Visualizations
# Plot 1: Completion Rate by Phase
plt.figure(figsize=(10, 6))
sns.barplot(x=phase_completion.index, y=phase_completion.values, palette='Blues_d')
plt.title('Trial Completion Rate by Phase')
plt.ylabel('Completion Rate (Proportion)')
plt.xticks(rotation=45)
plt.tight_layout()
plt.savefig('completion_rate_by_phase.png')

# Plot 2: Completion Rate by Funder
plt.figure(figsize=(10, 6))
sns.barplot(x=funder_completion.index, y=funder_completion.values, palette='Greens_d')
plt.title('Trial Completion Rate by Funder Category')
plt.ylabel('Completion Rate (Proportion)')
plt.tight_layout()
plt.savefig('completion_rate_by_funder.png')

# Plot 3: Median Enrollment by Status
plt.figure(figsize=(10, 6))
sns.barplot(x=enrollment_comparison.index, y=enrollment_comparison['median'], palette='Oranges_d')
plt.title('Median Enrollment Size by Trial Status')
plt.ylabel('Median Enrollment')
plt.tight_layout()
plt.savefig('enrollment_by_status.png')

print("Phase Completion Rates:\n", phase_completion)
print("\nFunder Completion Rates:\n", funder_completion)
print("\nEnrollment Stats by Status:\n", enrollment_comparison)
print("\nPhase Distribution of Terminated/Withdrawn:\n", tw_phase_dist)
```