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

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

# Preprocessing
df['Enrollment'] = pd.to_numeric(df['Enrollment'], errors='coerce')
df['Start Date'] = pd.to_datetime(df['Start Date'], errors='coerce')
df['Completion Date'] = pd.to_datetime(df['Completion Date'], errors='coerce')
df['Duration_Days'] = (df['Completion Date'] - df['Start Date']).dt.days
df['Duration_Days'] = df['Duration_Days'].apply(lambda x: x if (x > 0 and x < 2000) else np.nan) # Clean outliers
df['Is_Completed'] = (df['Status'] == 'Completed').astype(int)

def simplify_funder(funder_str):
    if pd.isna(funder_str): return 'Other'
    if 'Industry' in funder_str: return 'Industry'
    if 'NIH' in funder_str: return 'NIH'
    if 'U.S. Fed' in funder_str: return 'U.S. Fed'
    return 'Other'
df['Funder_Category'] = df['Funded Bys'].apply(simplify_funder)

# 1. Normality Testing (KS Test)
# Null Hypothesis: Data is normally distributed.
ks_enrollment = stats.kstest(df['Enrollment'].dropna(), 'norm')
ks_duration = stats.kstest(df['Duration_Days'].dropna(), 'norm')

# 2. Correlation Analysis (Spearman)
# Spearman is used because variables are highly skewed and Phase is ordinal.
phase_map = {'Early Phase 1': 0, 'Phase 1': 1, 'Phase 1|Phase 2': 1.5, 'Phase 2': 2, 'Phase 2|Phase 3': 2.5, 'Phase 3': 3, 'Phase 4': 4}
df['Phase_Ordinal'] = df['Phases'].map(phase_map)
corr_spearman = df[['Enrollment', 'Duration_Days', 'Phase_Ordinal', 'Is_Completed']].corr(method='spearman')

# 3. Group Comparison (Kruskal-Wallis)
# Does Enrollment vary by Funder Category?
funder_groups = [group['Enrollment'].dropna() for name, group in df.groupby('Funder_Category')]
h_stat, p_kruskal = stats.kruskal(*funder_groups)
```

```
# 4. Hypothesis Testing (Chi-Square)
# Association between Phase and Completion Status (excluding Missing phases)
df_phase_comp = df.dropna(subset=['Phases'])
contingency_phase = pd.crosstab(df_phase_comp['Phases'], df_phase_comp['Is_Completed'])
chi2_phase, p_chi2_phase, _, _ = stats.chi2_contingency(contingency_phase)

# Visualizations
plt.figure(figsize=(10, 6))
sns.heatmap(corr_spearman, annot=True, cmap='YlGnBu', fmt=".2f")
plt.title('Spearman Correlation Heatmap (Non-Parametric)')
plt.tight_layout()
plt.savefig('spearman_correlation.png')

# Distribution of Enrollment (Log scale) to show skewness
plt.figure(figsize=(10, 6))
sns.histplot(df['Enrollment'].dropna(), kde=True, log_scale=True, color='purple')
plt.title('Distribution of Trial Enrollment (Log Scale)')
plt.xlabel('Enrollment (Log)')
plt.savefig('enrollment_distribution.png')

print("KS Test p-values (Normality): Enrollment =", ks_enrollment.pvalue, ", Duration =", ks_duration.pvalue)
print("Spearman Correlation Matrix:\n", corr_spearman)
print("\nKruskal-Wallis (Enrollment by Funder): p-value =", p_kruskal)
print("Chi-Square (Phase vs Completion): p-value =", p_chi2_phase)
```