```
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')

# Convert dates to datetime
df['Start Date'] = pd.to_datetime(df['Start Date'], errors='coerce')
df['Completion Date'] = pd.to_datetime(df['Completion Date'], errors='coerce')

# Calculate duration in days
df['Duration_Days'] = (df['Completion Date'] - df['Start Date']).dt.days

# Filter out rows with missing or non-positive duration
df_duration = df[df['Duration_Days'] > 0].copy()

# Standardize Phases
# Group combined phases into their higher level for broader analysis or keep as is
valid_phases = ['Phase 1', 'Phase 2', 'Phase 3', 'Phase 4', 'Phase 1|Phase 2', 'Phase 2|Phase 3', 'Early Phase 1']
df_phase_dur = df_duration[df_duration['Phases'].isin(valid_phases)]

# Typical duration by Phase
phase_duration_stats = df_phase_dur.groupby('Phases')['Duration_Days'].agg(['median', 'mean', 'count']).sort_values(by='median')

# Standardize Conditions
covid_variants = ['COVID-19', 'Covid19', 'COVID', 'Covid-19', 'SARS-CoV-2', 'Coronavirus Infection',
                  'Coronavirus', 'COVID 19', 'COVID19', 'Corona Virus Infection', 'SARS-CoV2', 'SARS-CoV 2']

def get_primary_condition(cond):
    if pd.isna(cond): return 'Other'
    c_list = [c.strip() for c in cond.split('|')]
    for c in c_list:
        if c in covid_variants: return 'COVID-19'
    return c_list[0]

df_duration['Primary_Condition'] = df_duration['Conditions'].apply(get_primary_condition)
```

```
top_conds = df_duration['Primary_Condition'].value_counts().head(10).index
cond_duration_stats =
df_duration[df_duration['Primary_Condition'].isin(top_conds)].groupby('Primary_
Condition')['Duration_Days'].agg(['median', 'mean',
'count']).sort_values(by='median')

# Outliers detection: Trials taking significantly longer than expected (e.g. >
90th percentile)
threshold = df_duration['Duration_Days'].quantile(0.95)
outliers = df_duration[df_duration['Duration_Days'] >
threshold].sort_values(by='Duration_Days', ascending=False)

# Visualizations
# 1. Duration by Phase
plt.figure(figsize=(12, 6))
sns.boxplot(data=df_phase_dur, x='Phases', y='Duration_Days', palette='Set3',
showfliers=False)
plt.title('Clinical Trial Duration by Phase (Excl. Outliers)')
plt.ylabel('Duration (Days)')
plt.xticks(rotation=45)
plt.tight_layout()
plt.savefig('duration_by_phase.png')

# 2. Duration by Condition
plt.figure(figsize=(12, 6))
sns.barplot(x=cond_duration_stats['median'], y=cond_duration_stats.index,
palette='coolwarm')
plt.title('Median Trial Duration by Top Therapeutic Areas')
plt.xlabel('Median Duration (Days)')
plt.tight_layout()
plt.savefig('duration_by_condition.png')

print("Phase Duration Stats (Days):")
print(phase_duration_stats)
print("\nTop Condition Duration Stats (Days):")
print(cond_duration_stats)
print(f"\n95th Percentile Threshold for Duration: {threshold} days")
print(f"Number of 'Long' Outlier Trials: {len(outliers)}")
print("\nSample of Longest Trials:")
print(outliers[['NCT Number', 'Title', 'Duration_Days']].head(5))
```