```
# -*- coding: utf-8 -*-
"""Project-2.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1QfL423HnK4xiTl1BtbEyOXIGNcTOE1a2
"""

#Sai Kiran Mangali
import pandas as pd
df = pd.read_csv('/content/world_development_data.csv')
descriptive_stats = df.describe()
#print(descriptive_stats)

missing_values = df.isnull().sum()
df = df.dropna()
import seaborn as sns
import matplotlib.pyplot as plt
sns.boxplot(x='Region', y='GDP', data=df)
plt.title('Boxplot of GDP by Region')
plt.show()

from scipy.stats import ttest_ind
region_A = df[df['Region'] == 'Asia']['GDP'].dropna()
region_B = df[df['Region'] == 'Americas']['GDP'].dropna()

if region_A.isnull().any() or region_B.isnull().any():
    print("Warning: Missing values found.")
else:
    t_stat, p_value = ttest_ind(region_A, region_B)
    print(f'T-statistic: {t_stat}, P-value: {p_value}')

from scipy.stats import ttest_rel

before_policy = df[df['Year'] == 2000]['GDP']
after_policy = df[df['Year'] == 2020]['GDP']

t_stat, p_value = ttest_rel(before_policy, after_policy)
print(f'T-statistic: {t_stat}, P-value: {p_value}')

from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error

features = ['PopDens', 'FertRate']
target = 'GDP'
X_train, X_test, y_train, y_test = train_test_split(df[features], df[target],
test_size=0.2, random_state=42)

model = LinearRegression()
model.fit(X_train, y_train)
predictions = model.predict(X_test)

mse = mean_squared_error(y_test, predictions)
print(f'Mean Squared Error: {mse}')

from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
```

```
features_for_clustering = ['PopDens', 'GDP', 'LifeExpBirth']
scaler = StandardScaler()
df_scaled = scaler.fit_transform(df[features_for_clustering])
kmeans_model = KMeans(n_clusters=3, random_state=42)
df['Cluster'] = kmeans_model.fit_predict(df_scaled)
cluster_means = df.groupby('Cluster').mean()
print(cluster_means)
```