Plagiarism and AI Content Detection Report

import pandas as pd .txt

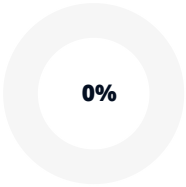
Scan details

Scan time:
January 7th, 2024 at 13:48 UTC

Total Pages:
1

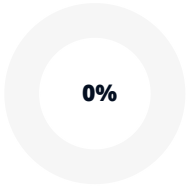
Total Words:
56

Plagiarism Detection



Types of plagiarism		Words
Identical	0%	0
Minor Changes	0%	0
Paparaphrased	0%	0
Omitted Words	0%	0

AI Content Detection



Text coverage		Words
AI text	0%	0
Human text	100%	56

Learn more

Plagiarism Results: No results found!

Plagiarism Report Content

```
import pandas as pd
import matplotlib.pyplot as plt
weatherp=pd.read_csv('weathers.csv')
des_status=weatherp[['MinTemp','MaxTemp','Rainfall','Evaporation']].describe()
print(des_status)
plt.figure(figsize=(12,6))
plt.plot(weatherp.index,weatherp['MinTemp'],label='MinTemp')
plt.plot(weatherp.index,weatherp['MaxTemp'],label='MaxTemp')
plt.plot(weatherp.index,weatherp['Rainfall'],label='Rainfall')
plt.plot(weatherp.index,weatherp['Evaporation'],label='Evaporation')
plt.title('Weather Variables Over Time')
plt.xlabe('Index')
ply.ylabel('Value')
plt.legend()
plt.show()

correlation=weatherp[['MinTemp','MaxTemp','Rainfall','Evaporation']].corr()
plt.figure(figsize=(8,6))
plt.imshow(correlation,cmap='coolwarm',interpolation='none')
plt.colorbar()
plt.xticks(range(len(correlation)),correlation.columns,rotation=45)
plt.yticks(range(len(correlation)),correlation.columns)
plt.title('Correlation Matrix')
plt.show()

plt.figure(figsize=(10,6))
plt.hist(weatherp['Rainfall'], bin=20, edgecolor='black')
plt.title('Rainfall Distribution')
plt.xlabel('Rainfall')
plt.ylabel('Frequency')
plt.show()

weatherp['Month']=pd.to_datetime(weatherp.index).month
seasonal=weatherp.groupby('Month')[['MinTemp','MaxTemp','Rainfall','Evaporation']].mean()
seasons=['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov','Dec']

plt.figure(figsize=(12,6))
plt.bar(seasons,seasonal['Rainfall'])
plt.title('Average Rainfall Across Different seasons')
plt.xlabel('MOnth')
plt.ylabel('Average Rainfall')
plt.show()
```