EDS Theory Activity No. 1

Name: Vansh Goyal

Subject: EDS

PRN: 202401070208

Roll No: ET1-69

Batch: ET1

```
import pandas as pd
import numpy as np
from datetime import datetime, timedelta
import random
```

```
# Generate random dates
def random_date(start, end):
    return start + timedelta(seconds=random.randint(0, int((end - start).total_seconds(

start_date = datetime(2000, 1, 1)
end_date = datetime(2002, 12, 31)

# Dummy dataset
data = {
    'From': [f'user{i}@enron.com' for i in range(1, 31)],
    'To': [f'user{random.randint(1, 30)}@enron.com' for _ in range(30)],
    'Subject': [random.choice(['Meeting', 'Report', 'Confidential Deal', None, 'Project
    'Body': [random.choice(['This is confidential', 'Please review', 'Monthly update',
    'Date': [random_date(start_date, end_date) for _ in range(30)],
    'Has_Attachment': [random.choice([True, False]) for _ in range(30)]
}

df = pd.DataFrame(data)
df.head()
```

	From	To	Subject	Body	Date	Has_Attachment
0	user1@enron.com	user9@enron.com	Report	Client discussion	2000-07-23 12:01:36	False
1	user2@enron.com	user23@enron.com	Confidential Deal	Monthly update	2002-01-12 01:34:40	True
2	user3@enron.com	user29@enron.com	Meeting	Please review	2000-10-08 13:57:21	False
3	user4@enron.com	user28@enron.com	Meeting	Client discussion	2000-09-01 05:40:29	True
4	user5@enron.com	user21@enron.com	Meeting	Client discussion	2000-10-07 22:05:03	False

```
total_emails = df.shape[0]
total_emails
```

```
30
```

```
[29] unique_senders = df['From'].nunique()
     unique_senders
```

```
30
```

```
[30] top_senders = df['From'].value_counts().head(5)
     top_senders
```

From	count
user1@enron.com	1
user2@enron.com	1
user3@enron.com	1
user4@enron.com	1
user5@enron.com	1

dtype: int64

```
[31] avg_subject_length = df['Subject'].dropna().apply(len).mean()
     avg_subject_length
```

```
np.float64(10.72)
```

```
     empty_subjects = df['Subject'].isna().sum()
     empty_subjects
```

```
np.int64(5)
```

```
[33] top_recipient = df['To'].value_counts().idxmax()
     top_recipient
```

```
'user28@enron.com'
```

```
[34] df['Date'] = pd.to_datetime(df['Date'], errors='coerce')
     emails_2001 = df[df['Date'].dt.year == 2001].shape[0]
     emails_2001
```

```
9
```

```
[35] median_body_length = df['Body'].dropna().apply(len).median()
     median_body_length
```

```
17.0
```

```
[36] most_active_month = df['Date'].dt.month.value_counts().idxmax()
     most_active_month
```

```
np.int32(1)
```

```
[37] confidential_emails = df[
         df['Subject'].fillna('').str.contains('confidential', case=False) |
         df['Body'].fillna('').str.contains('confidential', case=False)
     ].shape[0]
     confidential_emails
```

```
12
```

```
[38] self_emails = df[df['From'] == df['To']]['From'].unique()
     self_emails
```

```
array(['user8@enron.com', 'user19@enron.com', 'user28@enron.com'],
      dtype=object)
```

```
[39] earliest_email = df['Date'].min()
     earliest_email
```

```
Timestamp('2000-01-08 12:18:44')
```

```
[40] latest_email = df['Date'].max()
     latest_email
```

```
Timestamp('2002-12-13 20:20:43')
```

```
[41] attachment_percentage = (df['Has_Attachment'].sum() / len(df)) * 100
     attachment_percentage
```

```
np.float64(63.33333333333333)
```

```
[42] top_subjects = df['Subject'].value_counts().head(3)
     top_subjects
```

Subject	count
Meeting	12
Confidential Deal	6
Project Update	5

dtype: int64

```
[43] senders = set(df['From'].dropna())
     receivers = set(df['To'].dropna())
     only_receivers = receivers - senders
     only_receivers
```

```
set()
```

```
[44] most_common_day = df['Date'].dt.day_name().value_counts().idxmax()
     most_common_day
```

```
'Thursday'
```

```
[45] emails_by_sender = df.groupby('From').size()
     emails_by_sender
```

From	
user10@enron.com	1
user11@enron.com	1
user12@enron.com	1
user13@enron.com	1
user14@enron.com	1
user15@enron.com	1
user16@enron.com	1
user17@enron.com	1
user18@enron.com	1
user19@enron.com	1
user1@enron.com	1
user20@enron.com	1
user21@enron.com	1
user22@enron.com	1
user23@enron.com	1
user24@enron.com	1
user25@enron.com	1
user26@enron.com	1
user27@enron.com	1
user28@enron.com	1
user29@enron.com	1
user2@enron.com	1
user30@enron.com	1
user3@enron.com	1
user4@enron.com	1
user5@enron.com	1
user6@enron.com	1
user7@enron.com	1
user8@enron.com	1

user2@enron.com	1
user30@enron.com	1
user3@enron.com	1
user4@enron.com	1
user5@enron.com	1
user6@enron.com	1
user7@enron.com	1
user8@enron.com	1
user9@enron.com	1

dtype: int64

[46]
```
df['Subject_Length'] = df['Subject'].fillna('').apply(len)
df['Body_Length'] = df['Body'].fillna('').apply(len)
correlation = df[['Subject_Length', 'Body_Length']].corr().iloc[0,1]
correlation
```

```
np.float64(-0.19202434719162734)
```

[47]
```
s_outside_hours = df[(df['Date'].dt.hour < 9) | (df['Date'].dt.hour > 17)].shape[0]
s_outside_hours
```

```
20
```