

Instruksi dan Rubrik Penilaian Tugas Mandiri Praktik Function

(Sesi: Dataframe I: Sorting, Filtering, and Add Column)

Nama : Filbert Leonardo

Kelompok : 3

1. 5 nama penumpang perempuan dengan umur termuda serta tahun kelahirannya.

```
# Menghitung usia dengan mengurangkan tahun kelahiran dari tahun yang diberikan
year = 2024
df['age'] = year - df['payCardBirthDate']

df.head()
```

InStops	tapInStopsName	tapInStopsLat	tapInStopsLon	stopStartSeq	tapInTime	tapOutStops	tapOutStopsName	tapOutStopsLat	tapOutStopsLon	stopEndSeq	tapOutTime	payAmount	age
P00142	Pal Putih	-6.184631	106.84402	7	2023-04-03 05:21:44	P00253	Tegalan	-6.203101	106.85715	12.0	2023-04-03 06:00:53	3500.0	16
301963P	Kemenkes 2	-6.228700	106.83302	13	2023-04-03 05:42:44	B03307P	Sampoerna Strategic	-6.217152	106.81892	21.0	2023-04-03 06:40:01	3500.0	27
300499P	Gg. Kunir II	-6.133132	106.81435	38	2023-04-03 05:59:06	B04962P	Simpang Kunir Kemukus	-6.133731	106.81475	39.0	2023-04-03 06:50:55	3500.0	32
305587P	Taman Elok 1	-6.195743	106.93526	23	2023-04-03 05:44:51	B03090P	Raya Penggilingan	-6.183068	106.93194	29.0	2023-04-03 06:28:16	3500.0	46
P00239	Sunter Boulevard Barat	-6.149650	106.88900	5	2023-04-03 06:17:35	P00098	Kali Besar Barat	-6.135355	106.81143	15.0	2023-04-03 06:57:03	3500.0	42

```
# Filter data untuk penumpang perempuan
df_perempuan = df[df['payCardSex'] == 'F']

# Urutkan berdasarkan tanggal lahir secara menurun
df_perempuan = df_perempuan.sort_values('payCardBirthDate', ascending=False)

# Ambil lima data pertama
df_perempuan = df_perempuan[['payCardName', 'payCardBirthDate', 'age']].head(5)

# Ganti nama kolom
df_perempuan = df_perempuan.rename(columns={'payCardName': 'namaPenumpang', 'payCardBirthDate': 'tahunKelahiran', 'age': 'umur'})

# Tampilkan DataFrame hasilnya
df_perempuan
```

	namaPenumpang	tahunKelahiran	umur
4106	R. Lili Permata, S.E.	2012	12
20322	Catur Najmudin	2012	12
36527	Rahmi Saptono	2012	12
10122	Puspa Sihombing	2012	12
14649	Sutan Nasim Saragih	2012	12

2. **Jumlah transaksi** yang dilakukan oleh penumpang yang sudah pensiun (>55 tahun) serta berdasarkan **jenis kelaminnya** masing-masing.

```
[13] # Filter DataFrame berdasarkan usia lebih dari 55 tahun
df_retire = df.query('age > 55')
total_transaksi_retire = df_retire['payAmount'].shape[0]
print("Banyak Transaksi Penumpang yang Sudah Pensiun (>55 tahun):", total_transaksi_retire)

# Kelompokkan berdasarkan jenis kelamin dan banyaknya data
df_retire = df_retire.groupby('payCardSex')['payAmount'].size().reset_index()

# Ubah kode 'F' menjadi 'Perempuan' dan 'M' menjadi 'Laki-Laki'
df_retire['payCardSex'] = df_retire['payCardSex'].replace({'F': 'Perempuan', 'M': 'Laki-Laki'})

# Ubah nama kolom
df_retire = df_retire.rename(columns={'payCardSex': 'Jenis Kelamin', 'payAmount': 'Banyak Transaksi'})

# Tampilkan DataFrame hasilnya
print("\nBanyak Transaksi Penumpang yang Sudah Pensiun Berdasarkan Jenis Kelamin:")
df_retire
```

Jumlah Transaksi Penumpang yang Sudah Pensiun (>55 tahun): 2438

Jumlah Transaksi Penumpang yang Sudah Pensiun Berdasarkan Jenis Kelamin:

	Jenis Kelamin	Jumlah Transaksi
0	Perempuan	655
1	Laki-Laki	1783

3. Detail transaksi pertama berdasarkan *tapInTime* dengan fungsi **iloc**

```
# Mengurutkan DataFrame berdasarkan kolom 'tapInTime'
df_transaksi = df.sort_values('tapInTime')

# Mengambil baris pertama dari DataFrame yang telah diurutkan
df_transaksi = df_transaksi.iloc[0].reset_index()

# Menampilkan DataFrame
df_transaksi
```

	index	37751
0	transID	LDEA875J4U32YZ
1	payCardID	213107623841273
2	payCardBank	emoney
3	payCardName	Natalia Pratama
4	payCardSex	M
5	payCardBirthDate	1964
6	corridorID	11D
7	corridorName	Pulo Gebang - Pulo Gadung 2 via PIK
8	direction	0.0
9	tapInStops	B05823P
10	tapInStopsName	United Tractors 1
11	tapInStopsLat	-6.18326
12	tapInStopsLon	106.93243
13	stopStartSeq	28
14	tapInTime	2023-04-01 06:22:27

4. **Jumlah Transaksi Gratis** yang bernilai 0 rupiah serta informasi mengenai **penumpang** dan **rute perjalanannya** berdasarkan **jumlah pemberhentian terbanyak**.

```
[133] # Filter data untuk transaksi gratis
df_free = df[df['payAmount'] == 0]
print('Jumlah Transaksi Gratis:', df_free.shape[0])

# Hitung jumlah berhenti untuk setiap transaksi
df['stopCount'] = df['stopEndSeq'] - df['stopStartSeq']

# Temukan transaksi dengan jumlah berhenti maksimum
df_maxStop = df[df['stopCount'] == df['stopCount'].max()]

# Tampilkan kolom-kolom yang relevan untuk transaksi dengan jumlah berhenti maksimum
df_maxStop_info = df_maxStop[['payCardName', 'corridorName', 'tapInStopsName', 'tapOutStopsName', 'payAmount', 'stopCount']]

# Tampilkan informasi transaksi dengan jumlah berhenti maksimum
print("Informasi Transaksi dengan Jumlah Berhenti Maksimum:")
df_maxStop_info
```

Jumlah Transaksi Gratis: 16648
Informasi Transaksi dengan Jumlah Berhenti Maksimum:

	payCardName	corridorName	tapInStopsName	tapOutStopsName	payAmount	stopCount
	34447	Karen Pertiwi	Cilincing - Rorotan	RS Pelabuhan Jakarta	0.0	64.0

5. **Biaya yang dikeluarkan dan rute perjalanan** untuk transaksi dengan **jarak tempuh** **terlama** dan **jumlah pemberhentian terbanyak**.

```
[28] # Mengubah kolom tapInTime dan tapOutTime menjadi tipe datetime
df['tapInTime'] = pd.to_datetime(df['tapInTime'])
df['tapOutTime'] = pd.to_datetime(df['tapOutTime'])

# Menghitung durasi perjalanan (menit)
df['duration'] = (df['tapOutTime'] - df['tapInTime']).dt.seconds / 60

# Memilih baris dengan durasi terpanjang
df_longest_duration = df[df['duration'] == df['duration'].max()].iloc[:1]
# Memilih baris dengan jumlah berhenti terbanyak
df_result = df_longest_duration[df_longest_duration['stopCount'] == df_longest_duration['stopCount'].max()].iloc[:1]

print("Data dengan jarak tempuh terlama dan jumlah pemberhentian terbanyak:")
print("Nama Penumpang:", df_result['payCardName'].values[0],
      "\nKoridor:", df_result['corridorName'].values[0],
      "\nRute Perjalanan:", df_result['tapInStopsName'].values[0], "-", df_result['tapOutStopsName'].values[0],
      "\nBiaya:", df_result['payAmount'].values[0], "rupiah",
      "\nJarak tempuh terlama:", df_result['duration'].values[0], "menit")

df_result[['payCardName', 'corridorName', 'tapInStopsName', 'tapOutStopsName', 'payAmount', 'stopCount', 'tapInTime', 'tapOutTime', 'duration']]
```

Data dengan jarak tempuh terlama dan jumlah pemberhentian terbanyak:
Nama Penumpang: Langgeng Handayani
Koridor: Tanjung Priok - Senen via Taman BMW
Rute Perjalanan: Sbr. Jln. H. Ung - Jembatan Item
Biaya: 3500.0 rupiah
Jarak tempuh terlama: 179.05 menit

	payCardName	corridorName	tapInStopsName	tapOutStopsName	payAmount	stopCount	tapInTime	tapOutTime	duration
	37616	Langgeng Handayani	Tanjung Priok - Senen via Taman BMW	Sbr. Jln. H. Ung	3500.0	12.0	2023-04-23 05:36:41	2023-04-23 08:35:44	179.05

Project Folder:

[https://drive.google.com/drive/folders/1OpZZoKuHtBqvMsqtxB0PZmREDoyfe_2u?usp=drive link](https://drive.google.com/drive/folders/1OpZZoKuHtBqvMsqtxB0PZmREDoyfe_2u?usp=drive_link)