This document describes how to load the provided dataset and inspect its structure. Detailed descriptions for each data item will be provided in a separate reference document.

@since 2024.02.05

In [1]:
```
import os
import numpy as np
import pandas as pd
```

In [2]:
```
# Note: Set the file path associated with the dataset.
challenge2025_dataset_path = "{Dataset file path ...}"  # <=== Modify as appropriate.
```

Data items

- mACStatus
- mActivity
- mAmbience
- mBle
- mGps
- mLight
- mScreenStatus
- mUsageStats
- mWifi
- wHr
- wLight
- wPedo

In [3]:
```
# To print the list of data items
print("challenge 2025 dataset " + "="*5)
for file_name in sorted(os.listdir(challenge2025_dataset_path)):
```

```
    if file_name.endswith('.parquet'):
        print(file_name)
```

```
challenge 2025 dataset =====
ch2025_mACStatus.parquet
ch2025_mActivity.parquet
ch2025_mAmbience.parquet
ch2025_mBle.parquet
ch2025_mGps.parquet
ch2025_mLight.parquet
ch2025_mScreenStatus.parquet
ch2025_mUsageStats.parquet
ch2025_mWifi.parquet
ch2025_wHr.parquet
ch2025_wLight.parquet
ch2025_wPedo.parquet
```

mACStatus

In [4]:
```
data_item = "mACStatus"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 939896 entries, 0 to 939895
Data columns (total 3 columns):
 #   Column      Non-Null Count   Dtype         
---  ------      --------------   -----         
 0   subject_id  939896 non-null  object        
 1   timestamp   939896 non-null  datetime64[ns]
 2   m_charging  939896 non-null  int64         
dtypes: datetime64[ns](1), int64(1), object(1)
memory usage: 21.5+ MB
```

Out[4]:

	subject_id	timestamp	m_charging
0	id01	2024-06-26 12:03:00	0
1	id01	2024-06-26 12:04:00	0
2	id01	2024-06-26 12:05:00	0
3	id01	2024-06-26 12:06:00	0
4	id01	2024-06-26 12:07:00	0

mActivity

- subject_id
- timestamp
- m_activity
  - 0: IN_VEHICLE
  - 1: ON_BICYCLE
  - 2: ON_FOOT
  - 3: STILL
  - 4: UNKNOWN
  - 5: TILTING
  - 7: WALKING
  - 8: RUNNING

In [5]:
```
data_item = "mActivity"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 961062 entries, 0 to 961061
Data columns (total 3 columns):
 #   Column      Non-Null Count   Dtype
---  ------      --------------   -----
 0   subject_id  961062 non-null  object
 1   timestamp   961062 non-null  datetime64[ns]
 2   m_activity  961062 non-null  int64
dtypes: datetime64[ns](1), int64(1), object(1)
memory usage: 22.0+ MB
```

Out[5]:

	subject_id	timestamp	m_activity
0	id01	2024-06-26 12:03:00	4
1	id01	2024-06-26 12:04:00	0
2	id01	2024-06-26 12:05:00	0
3	id01	2024-06-26 12:06:00	0
4	id01	2024-06-26 12:07:00	0

mAmbience

Audio-based labels detected on smartphones. Recorded once every 2 minutes.

- subject_id
- timestamp
- ambience_labels: List of the top 10 labels along with their respective probabilities.

In [6]:

```
data_item = "mAmbience"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 476577 entries, 0 to 476576
Data columns (total 3 columns):
 #   Column      Non-Null Count   Dtype
---  ------      --------------   -----
 0   subject_id  476577 non-null  object
 1   timestamp   476577 non-null  datetime64[ns]
 2   m_ambience  476577 non-null  object
dtypes: datetime64[ns](1), object(2)
memory usage: 10.9+ MB
```

Out[6]:

	subject_id	timestamp	m_ambience
0	id01	2024-06-26 13:00:10	[[Music, 0.30902618], [Vehicle, 0.081680894], ...
1	id01	2024-06-26 13:02:10	[[Music, 0.62307084], [Vehicle, 0.021118319], ...
2	id01	2024-06-26 13:04:10	[[Horse, 0.25209898], [Animal, 0.24263993], [C...
3	id01	2024-06-26 13:06:10	[[Speech, 0.93433166], [Inside, large room or ...
4	id01	2024-06-26 13:08:10	[[Speech, 0.8935082], [Inside, small room, 0.0...

mBle

In [7]:
```
data_item = "mBle"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21830 entries, 0 to 21829
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   subject_id  21830 non-null  object
 1   timestamp   21830 non-null  datetime64[ns]
 2   m_ble       21830 non-null  object
dtypes: datetime64[ns](1), object(2)
memory usage: 511.8+ KB
```

Out[7]:

	subject_id	timestamp	m_ble
0	id01	2024-06-26 12:13:00	[{'address': '00:15:7C:11:80:8D', 'device_clas...
1	id01	2024-06-26 12:23:00	[{'address': '0A:B1:26:4D:76:21', 'device_clas...
2	id01	2024-06-26 12:33:00	[{'address': '04:F5:AE:39:95:E0', 'device_clas...
3	id01	2024-06-26 13:23:00	[{'address': '06:C0:D2:6D:9F:69', 'device_clas...
4	id01	2024-06-26 14:23:00	[{'address': '10:2B:41:74:9F:B1', 'device_clas...

mGps

GPS coordinate information generated by smartphones (with latitude and longitude converted to relative coordinates for privacy protection). Measured up to 12 times per minute.

- subject_id
- timestamp
- altitude
- latitude
- longitude
- speed

Note: Data for which both latitude and longitude are recorded as -1 were excluded.

In [8]:
```
data_item = "mGps"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 800611 entries, 0 to 800610
Data columns (total 3 columns):
 #   Column      Non-Null Count   Dtype
---  ------      --------------   -----
 0   subject_id  800611 non-null  object
 1   timestamp   800611 non-null  datetime64[ns]
 2   m_gps       800611 non-null  object
dtypes: datetime64[ns](1), object(2)
memory usage: 18.3+ MB
```

Out[8]:

	subject_id	timestamp	m_gps
0	id01	2024-06-26 12:03:00	[{'altitude': 110.6, 'latitude': 0.2077385, 'l...
1	id01	2024-06-26 12:04:00	[{'altitude': 110.8, 'latitude': 0.2078068, 'l...
2	id01	2024-06-26 12:05:00	[{'altitude': 110.7, 'latitude': 0.2078214, 'l...
3	id01	2024-06-26 12:06:00	[{'altitude': 110.7, 'latitude': 0.2078395, 'l...
4	id01	2024-06-26 12:07:00	[{'altitude': 110.8, 'latitude': 0.2078478, 'l...

mLight

Measured at 10-minute intervals.

- subject_id
- timestamp
- m_light: Ambient light in lx unit.

In [9]:
```
data_item = "mLight"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 96258 entries, 0 to 96257
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   subject_id  96258 non-null  object
 1   timestamp   96258 non-null  datetime64[ns]
 2   m_light     96258 non-null  float64
dtypes: datetime64[ns](1), float64(1), object(1)
memory usage: 2.2+ MB
```

Out[9]:

	subject_id	timestamp	m_light
0	id01	2024-06-26 12:03:00	534.0
1	id01	2024-06-26 12:13:00	846.0
2	id01	2024-06-26 12:23:00	826.0
3	id01	2024-06-26 12:33:00	851.0
4	id01	2024-06-26 12:43:00	428.0

mScreenStatus

In [10]:
```
data_item = "mScreenStatus"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 939653 entries, 0 to 939652
Data columns (total 3 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   subject_id    939653 non-null  object
 1   timestamp     939653 non-null  datetime64[ns]
 2   m_screen_use  939653 non-null  int64
dtypes: datetime64[ns](1), int64(1), object(1)
memory usage: 21.5+ MB
```

Out[10]:

	subject_id	timestamp	m_screen_use
0	id01	2024-06-26 12:03:00	0
1	id01	2024-06-26 12:04:00	0
2	id01	2024-06-26 12:05:00	0
3	id01	2024-06-26 12:06:00	0
4	id01	2024-06-26 12:07:00	0

mUsageStats

Smartphone app usage information, measured at 10-minute intervals.

- subject_id
- timestamp
- m_usage_stats: List of app names and their respective usage times.

In [11]:
```
data_item = "mUsageStats"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45197 entries, 0 to 45196
Data columns (total 3 columns):
 #   Column         Non-Null Count  Dtype         
---  ------         --------------  -----         
 0   subject_id     45197 non-null  object        
 1   timestamp      45197 non-null  datetime64[ns]
 2   m_usage_stats  45197 non-null  object        
dtypes: datetime64[ns](1), object(2)
memory usage: 1.0+ MB
```

Out[11]:

	subject_id	timestamp	m_usage_stats
0	id01	2024-06-26 13:00:00	[{'app_name': ' 캐시워크', 'total_time': 69}, {'ap...
1	id01	2024-06-26 13:10:00	[{'app_name': '통화', 'total_time': 26419}, {'ap...
2	id01	2024-06-26 13:20:00	[{'app_name': '메시지', 'total_time': 388651}, {'...
3	id01	2024-06-26 13:30:00	[{'app_name': '메시지', 'total_time': 211633}, {'...
4	id01	2024-06-26 13:50:00	[{'app_name': '카카오톡', 'total_time': 35446}, {'...

mWifi

In [12]:
```
data_item = "mWifi"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 76336 entries, 0 to 76335
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype         
---  ------      --------------  -----         
 0   subject_id  76336 non-null  object        
 1   timestamp   76336 non-null  datetime64[ns]
 2   m_wifi      76336 non-null  object        
dtypes: datetime64[ns](1), object(2)
memory usage: 1.7+ MB
```

Out[12]:

	subject_id	timestamp	m_wifi
0	id01	2024-06-26 12:03:00	[{'bssid': 'a0:0f:37:9a:5d:8b', 'rssi': -78}, ...
1	id01	2024-06-26 12:13:00	[{'bssid': 'a0:0f:37:9a:5d:8b', 'rssi': -79}, ...
2	id01	2024-06-26 12:23:00	[{'bssid': '10:e3:c7:0a:74:d1', 'rssi': -78}, ...
3	id01	2024-06-26 12:33:00	[{'bssid': '10:e3:c7:09:7f:bc', 'rssi': -80}, ...
4	id01	2024-06-26 12:43:00	[{'bssid': '56:46:ae:59:b1:13', 'rssi': -44}, ...

wHr

Heart rate data measured on a smartwatch (Galaxy Watch). Recorded up to 60 times per minute.

- subject_id
- timestamp
- heart_rate

In [13]:
```
data_item = "wHr"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 382918 entries, 0 to 382917
Data columns (total 3 columns):
 #   Column      Non-Null Count   Dtype         
---  ------      --------------   -----         
 0   subject_id  382918 non-null  object        
 1   timestamp   382918 non-null  datetime64[ns]
 2   heart_rate  382918 non-null  object        
dtypes: datetime64[ns](1), object(2)
memory usage: 8.8+ MB
```

Out[13]:

	subject_id	timestamp	heart_rate
0	id01	2024-06-26 12:23:00	[134, 134, 135, 133, 134, 135, 134, 135, 134, ...
1	id01	2024-06-26 12:24:00	[123, 122, 121, 120, 121, 121, 120, 118, 119, ...
2	id01	2024-06-26 12:25:00	[120, 119, 117, 116, 119, 121, 123, 123, 121, ...
3	id01	2024-06-26 12:26:00	[125, 124, 124, 124, 125, 124, 124, 123, 123, ...
4	id01	2024-06-26 12:27:00	[116, 116, 117, 118, 116, 116, 116, 117, 115, ...

wLight

Measured at 10-minute intervals.

- subject_id
- timestamp
- w_light: Ambient light in lx unit.

In [14]:
```
data_item = "wLight"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 633741 entries, 0 to 633740
Data columns (total 3 columns):
 #   Column      Non-Null Count   Dtype
---  ------      --------------   -----
 0   subject_id  633741 non-null  object
 1   timestamp   633741 non-null  datetime64[ns]
 2   w_light     633741 non-null  float64
dtypes: datetime64[ns](1), float64(1), object(1)
memory usage: 14.5+ MB
```

Out[14]:

	subject_id	timestamp	w_light
0	id01	2024-06-26 12:17:00	633.0
1	id01	2024-06-26 12:18:00	483.0
2	id01	2024-06-26 12:19:00	541.0
3	id01	2024-06-26 12:20:00	547.0
4	id01	2024-06-26 12:21:00	547.0

wPedo

Step count data and related information measured by the smartwatch.

- subject_id
- timestamp
- burned_calories
- distance
- speed
- steps
- step_frequency

In [15]:

```
data_item = "wPedo"
df_data = pd.read_parquet(os.path.join(challenge2025_dataset_path, f"ch2025_{data_item}.parquet"))
df_data.info()
df_data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 748100 entries, 0 to 748099
Data columns (total 9 columns):
 #   Column           Non-Null Count   Dtype
---  ------           --------------   -----
 0   subject_id       748100 non-null  object
 1   timestamp        748100 non-null  datetime64[ns]
 2   step             748100 non-null  int64
 3   step_frequency   748100 non-null  float64
 4   running_step     748100 non-null  int64
 5   walking_step     748100 non-null  int64
 6   distance         748100 non-null  float64
 7   speed            748100 non-null  float64
 8   burned_calories  748100 non-null  float64
dtypes: datetime64[ns](1), float64(4), int64(3), object(1)
memory usage: 51.4+ MB
```

Out[15]:

	subject_id	timestamp	step	step_frequency	running_step	walking_step	distance	speed	burned_calories
0	id01	2024-06-26 12:09:00	10	0.166667	0	0	8.33	0.138833	0.0
1	id01	2024-06-26 12:10:00	0	0.000000	0	0	0.00	0.000000	0.0
2	id01	2024-06-26 12:11:00	0	0.000000	0	0	0.00	0.000000	0.0
3	id01	2024-06-26 12:12:00	0	0.000000	0	0	0.00	0.000000	0.0
4	id01	2024-06-26 12:13:00	0	0.000000	0	0	0.00	0.000000	0.0

Six metrics (for 450 days)

Indicators of sleep quality, sleep health, fatigue, and stress levels. Each indicator is recorded as either 0, 1, or 2.

- subject_id
- sleep_date: Date of 'sleep record'.
- lifelog_date: Daily activity date (i.e., the day before 'sleep_date').
- Q1: Overall sleep quality
- Q2: Fatigue level

- Q3: Stress level
- S1: Total sleep time
- S2: Sleep efficiency
- S3: Sleep onset latency

In [16]:
```
file_name = "ch2025_metrics_train.csv"
df_label = pd.read_csv(os.path.join(challenge2025_dataset_path, file_name))
df_label.info()
df_label.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 450 entries, 0 to 449
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   subject_id    450 non-null    object
 1   sleep_date    450 non-null    object
 2   lifelog_date  450 non-null    object
 3   Q1            450 non-null    int64 
 4   Q2            450 non-null    int64 
 5   Q3            450 non-null    int64 
 6   S1            450 non-null    int64 
 7   S2            450 non-null    int64 
 8   S3            450 non-null    int64 
dtypes: int64(6), object(3)
memory usage: 31.8+ KB
```

Out[16]:

	subject_id	sleep_date	lifelog_date	Q1	Q2	Q3	S1	S2	S3
0	id01	2024-06-27	2024-06-26	0	0	0	0	0	1
1	id01	2024-06-28	2024-06-27	0	0	0	0	1	1
2	id01	2024-06-29	2024-06-28	1	0	0	1	1	1
3	id01	2024-06-30	2024-06-29	1	0	1	2	0	0
4	id01	2024-07-01	2024-06-30	0	1	1	1	1	1

Example submission format for prediction results.

For each of the 6 indicators, predicted values for 250 days should be recorded as either 0, 1, or 2.

- subject_id
- sleep_date
- lifelog_date
- Q1: Overall sleep quality
- Q2: Fatigue level
- Q3: Stress level
- S1: Total sleep time
- S2: Sleep efficiency
- S3: Sleep onset latency

In [17]:
```
file_name = "ch2025_submission_sample.csv"
df_label = pd.read_csv(os.path.join(challenge2025_dataset_path, file_name))
df_label.info()
df_label.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 250 entries, 0 to 249
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   subject_id    250 non-null    object
 1   sleep_date    250 non-null    object
 2   lifelog_date  250 non-null    object
 3   Q1            250 non-null    int64 
 4   Q2            250 non-null    int64 
 5   Q3            250 non-null    int64 
 6   S1            250 non-null    int64 
 7   S2            250 non-null    int64 
 8   S3            250 non-null    int64 
dtypes: int64(6), object(3)
memory usage: 17.7+ KB
```

Out[17]:

	subject_id	sleep_date	lifelog_date	Q1	Q2	Q3	S1	S2	S3
0	id01	2024-07-31	2024-07-30	0	0	0	0	0	0
1	id01	2024-08-01	2024-07-31	0	0	0	0	0	0
2	id01	2024-08-02	2024-08-01	0	0	0	0	0	0
3	id01	2024-08-03	2024-08-02	0	0	0	0	0	0
4	id01	2024-08-04	2024-08-03	0	0	0	0	0	0