

# ingest\_sensor\_metadata\_01

July 2, 2025

```
[1]: # CONFIG CELL
from notebook_utils import set_root_directory

set_root_directory()
```

```
[2]: # IMPORTS
from app.sensor_ingestor import SensorIngestor
```

```
[3]: # CONSTANTS
SENSOR_METADATA_FILE = "data/Metadane oraz kody stacji i stanowisk pomiarowych.
↳xlsx"
SENSOR_METADATA_RESULT = "input_files/sensor_metadata.parquet"
```

```
[6]: # CODE
ingestor = SensorIngestor(file_path=SENSOR_METADATA_FILE)
ingested_sensor_metadata = ingestor.transform()
ingested_sensor_metadata.to_parquet(SENSOR_METADATA_RESULT, index=False)
```

```
[7]: ingested_sensor_metadata.head(5)
```

```
[7]:
```

	station_code	station_type	area_type	station_kind	\
0	DsBialka	przemysłowa	podmiejski	kontenerowa	stacjonarna
1	DsBielGrot	tłó	miejski		w budynku
2	DsBogChop	przemysłowa	miejski	kontenerowa	stacjonarna
3	DsBogZatonieMob	przemysłowa	miejski		mobilna
4	DsBogatFrancMOB	tłó	miejski		mobilna

  

	latitude	longitude	province	city	sensor_id	old_station_code
0	51.197783	16.117390	DOLNOŚLĄSKIE	Białka	1	{}
1	50.682510	16.617348	DOLNOŚLĄSKIE	Bielawa	2	{}
2	50.905856	14.967175	DOLNOŚLĄSKIE	Bogatynia	4	{}
3	50.943245	14.913327	DOLNOŚLĄSKIE	Bogatynia	5	{}
4	50.940998	14.916790	DOLNOŚLĄSKIE	Bogatynia	3	{DsBogatMob}

```
[8]: SENSOR_TO_INVESTIAGE = "PkRzeszWIOSLang"
```

```
[9]: ingested_sensor_metadata[
```

```
    ingested_sensor_metadata["old_station_code"].apply(lambda x: □  
↳SENSOR_TO_INVESTIAGE in x)  
]
```

[9]: Empty DataFrame  
Columns: [station\_code, station\_type, area\_type, station\_kind, latitude, longitude, province, city, sensor\_id, old\_station\_code]  
Index: []

```
[10]: ingested_sensor_metadata.query(f"station_code == '{SENSOR_TO_INVESTIAGE}')
```

[10]: Empty DataFrame  
Columns: [station\_code, station\_type, area\_type, station\_kind, latitude, longitude, province, city, sensor\_id, old\_station\_code]  
Index: []