KARTA PRACY UCZESTNIKA

Moduł 2 Przygotowanie i łączenie danych

Kontrola jakości i udokumentowane decyzje w Jupyter Notebook

Czas	Pracujesz z	Rezultat
75–100 min	projektem z M1, Pandas i plikami CSV oraz XLSX	sprawdzonym zbiorem pomiary_clean.csv i dokumentacją decyzji

W module 1 zadbałeś o odtwarzalność pracy. Teraz sprawdzisz, czy same dane nadają się do dalszej analizy. Przy każdej zmianie zapisz, co wykryłeś, dlaczego podejmujesz daną decyzję i jak potwierdzasz jej skutek.

Cel i rezultat modułu

Przeprowadzisz audyt jakości, przygotujesz kopię roboczą pomiarów, połączysz ją z kartoteką urządzeń i sprawdzisz wynik po zapisie. Przygotowany zbiór stanie się punktem wyjścia do modułu 3.

- Zachowasz niezmienione źródła i odrębny notebook M1.
- Zapiszesz audyt przed i po przygotowaniu oraz dziennik decyzji.
- Uzupełnisz słownik o zmienione nazwy, typy i pola pochodne.

Zanim zaczniesz

Kontynuuj projekt eksploatacja_urzadzen z karty M1. W folderze notebooks utwórz M2_przygotowanie_danych.ipynb, przenosząc z M1 układ sekcji i konfigurację ścieżek. Obliczenia M2 zaczynaj od ponownego importu pełnego zbioru, a nie od podzbioru U-07 ani zmiennych z sesji M1.

Plik w projekcie	Źródło i przeznaczenie
data/raw/pomiary_eksploatacyjne.csv	Plik Dane/pomiary_eksploatacyjne.csv z pakietu. Główne pomiary, tylko do odczytu.
data/raw/urzadzenia.xlsx	Plik Dane/urzadzenia.xlsx z pakietu. Kartoteka w arkuszu urzadzenia.
data/raw/przyklady_kontrolowane_m2.csv	Skopiuj z Dane w pakiecie. Oddzielne ćwiczenie problemów jakości.
docs/slownik_danych.md	Kopia Dane/slownik_danych.md utworzona w M1. Uzupełniaj opis wyniku, zachowując opis źródeł.

Nie dopisuj przykładów kontrolowanych do głównych pomiarów. Nie korzystaj z rozwiązania referencyjnego ani klucza prowadzącego podczas wykonywania karty. Liczby w audycie i dzienniku uzyskaj samodzielnie.

Workflow pracy krok po kroku

Rozdział 1 Bezpieczne wczytywanie CSV i XLSX

Uruchom notebook z katalogiem roboczym notebooks. Sprawdź Path.cwd() i istnienie plików. Jeśli katalog jest inny, popraw punkt startu; nie zastępuj ścieżek przenośnych ścieżką do swojego konta.

```
from pathlib import Path
import pandas as pd

PROJECT_DIR = Path.cwd().parent
RAW_DIR = PROJECT_DIR / "data" / "raw"
PROCESSED_DIR = PROJECT_DIR / "data" / "processed"
TABLES_DIR = PROJECT_DIR / "outputs" / "tables"
plik_pomiary = RAW_DIR / "pomiary_eksploatacyjne.csv"
plik_urzadzenia = RAW_DIR / "urzadzenia.xlsx"
assert plik_pomiary.is_file() and plik_urzadzenia.is_file()

pomiary_raw = pd.read_csv(plik_pomiary, sep=",", encoding="utf-8")
with pd.ExcelFile(plik_urzadzenia) as skoroszyt:
    print(skoroszyt.sheet_names)
    urzadzenia_raw = pd.read_excel(skoroszyt, sheet_name="urzadzenia")
```

- Dla obu tabel obejrzyj head(), shape, columns oraz dtypes. Porównaj je ze słownikiem danych.
- W osobnej zmiennej wczytaj CSV z separatorem średnikowym. Porównaj strukturę i zapisz, dlaczego sam brak błędu nie potwierdza poprawności importu.
- Porównaj domyślny odczyt XLSX z jawnym wyborem arkusza urzadzenia. Arkusz opis zawiera objaśnienia, nie kartotekę.

Punkt kontrolny: zapisz w Markdown źródło, parametry, wykonane kontrole i werdykt dla każdego importu. Do dalszej pracy dopuszczasz wyłącznie prawidłowo wczytane tabele.

Rozdział 2 Pierwszy audyt zbioru

Zbuduj profil jakości przed modyfikacją danych. Oceniaj go względem znaczenia pól i reguł w docs/slownik_danych.md. Powtarzające się urządzenie jest naturalne w serii pomiarów; kluczem obserwacji jest para czas i urzadzenie.

- Policz braki i ich udział, pełne duplikaty oraz powtórzenia klucza czas + urzadzenie.
- Sprawdź zakres czasu, kategorie tryb i status oraz wartości liczbowe z jednostkami.
- W kartotece sprawdź brakujące i powtórzone identyfikatory. Porównaj zbiory identyfikatorów obu źródeł.
- Zapisz tabelę audytu z polami: kontrola, przed, po, interpretacja. Kolumnę po uzupełnisz po wykonaniu zmian.

Zastanów się: który wykryty problem mógłby zmienić późniejsze porównanie urządzeń? Zapisz jeden przykład i wskaż kontrolę, która go ujawnia.

Decyzje i standaryzacja danych

Rozdział 3 Braki duplikaty i identyfikatory

Utwórz pomiary_clean = pomiary_raw.copy() oraz urzadzenia_clean = urzadzenia_raw.copy(). Wszystkie poprawki wykonuj w kopiach. Dla każdego problemu rozważ pozostawienie wartości, poprawę na podstawie wiarygodnego źródła, oznaczenie braku albo usunięcie rekordu.

- Nie zastępuj braku energii zerem bez uzasadnienia. Zero jest pomiarem, a nie oznaczeniem niewiedzy.
- Pełny duplikat i konflikt klucza wymagają odrębnej oceny. Przy konflikcie obejrzyj wszystkie rekordy; nie wybieraj automatycznie pierwszego.
- Usuwaj rekord dopiero po zapisaniu reguły i liczby objętych nią wierszy. Jeśli problem dotyczy jednej komórki, oceń skutek usunięcia pozostałych pomiarów z tego wiersza.

W notebooku prowadź sekcję Dziennik decyzji dotyczących jakości danych. Każdy wpis powinien zawierać pole lub klucz rekordu, dowód, decyzję, uzasadnienie oraz kontrolę skutku.

Uzupełnij dla własnego przypadku: „Wykryłem … na podstawie … . Wybieram …, ponieważ … . Poprawność decyzji sprawdzam przez …”. Zapisuj także uzasadnione decyzje o pozostawieniu danych bez zmian.

Rozdział 4 Typy daty kategorie i jednostki

- Konwertuj czas według formatu źródła. Jeśli używasz errors="coerce", oddziel pierwotne braki od wartości, które dopiero po konwersji stały się NaT.
- Ujednolicaj zapis kategorii: usuń zbędne spacje, tryb zapisuj małymi literami, a status wielkimi. Po zmianie sprawdź wartości względem słownika, zanim nadasz typ category.
- Identyfikatory zachowaj jako tekst, bez utraty zer i łączników. Zastosuj te same reguły w pomiarach i kartotece.
- Dla produkcji użyj typu Int64 dopuszczającego brak. Nie zaokrąglaj wartości tylko po to, by konwersja się powiodła.

```
czas_po = pd.to_datetime(
    pomiary_clean["czas"], format="%Y-%m-%d %H:%M", errors="coerce"
)
blad_daty = pomiary_clean["czas"].notna() & czas_po.isna()
print(pomiary_clean.loc[blad_daty, ["czas", "urzadzenie"]])
# Rozstrzygnij wskazane przypadki przed przyjęciem konwersji.
```

Oddzielne ćwiczenie na przykładach kontrolowanych

Wczytaj przyklady_kontrolowane_m2.csv do osobnej zmiennej z sep=";", decimal=",", encoding="utf-8" i na_values=["N/D"]. Porównaj duplikat pełny z konfliktem klucza. Sprawdź różne formaty dat, kodowanie tekstu oraz jednostki temperatury i energii.

Przeliczaj wartości według kolumn jednostek, a daty według rozpoznanego formatu. Zachowaj wartość źródłową obok przeliczonej. Niepoprawnej daty nie zgaduj. Wynik ćwiczenia pozostaje osobno i nie trafia do pomiary_clean.csv.

Punkt kontrolny: po standaryzacji ponownie sprawdź klucz czas + urzadzenie. Ujednolicenie zapisu może ujawnić powtórzenia, których wcześniej nie było widać.

Ocena wartości i kontrolowane łączenie

Rozdział 5 Zakresy dopuszczalne i wartości odstające

Zapisz reguły ze słownika jako kontrole w kodzie. Oddziel naruszenie dopuszczalnego zakresu od przekroczenia progu diagnostycznego. Zakresy w tym kursie dotyczą syntetycznego zbioru i nie są uniwersalnymi normami eksploatacji urządzeń.

- Dla pól liczbowych oblicz kwartyle i granice IQR. Zapisz liczbę dostępnych pomiarów, granice oraz liczbę wskazań; brak nie jest wartością odstającą.
- Obejrzyj wskazane rekordy w kontekście urządzenia i sąsiednich pomiarów. Porównaj sygnał globalny z rozkładem dla danego urządzenia.
- Nadaj klasyfikację: błąd, do sprawdzenia albo prawidłowa wartość nietypowa. Sam sygnał IQR nie uzasadnia usunięcia.
- Udokumentuj zmianę lub pozostawienie wartości i sprawdź skutek. Nie poprawiaj danych po to, by zniknęły wszystkie sygnały jakości.

Punkt kontrolny: potrafisz wskazać źródło reguły, dowód dotyczący rekordu i powód swojej decyzji. Jeśli nie możesz ustalić przyczyny, zapisz ograniczenie zamiast zgadywać.

Rozdział 6 Transformacje łączenie i walidacja wyniku

W kopii kartoteki zmień nazwę typ na typ_urzadzenia. Dodaj rok_miesiac jako tekst YYYY-MM wyprowadzony z czasu. Przed połączeniem sprawdź kompletność kluczy, unikalność klucza kartoteki i różnice zbiorów identyfikatorów.

```
klucz = "urzadzenie"
assert pomiary_clean[klucz].notna().all()
assert urzadzenia_clean[klucz].notna().all()
assert urzadzenia_clean[klucz].is_unique
liczba_przed = len(pomiary_clean)
polaczone = pomiary_clean.merge(
    urzadzenia_clean, on=klucz, how="left",
    validate="many_to_one", indicator=True
)
print(polaczone["_merge"].value_counts())
assert len(polaczone) == liczba_przed
assert polaczone["_merge"].eq("both").all()
pomiary_clean = polaczone.drop(columns="_merge")
```

Wykonaj ten blok po przygotowaniu obu kopii. Gdy kontrola nie przechodzi, wróć do źródła problemu; nie wyłączaj validate. Przy left merge nie zobaczysz urządzeń występujących wyłącznie w kartotece — sprawdź je osobno przez różnicę zbiorów identyfikatorów.

Po połączeniu dodaj wiek_urzadzenia_dni: różnicę pomiędzy dniem pomiaru a datą instalacji, w pełnych dniach, jako Int64. Sprawdź brakujące daty i wartości ujemne. Zachowaj obie daty źródłowe.

Dla chętnych: zbuduj audit_data(df) albo porównaj dwie strategie obsługi jednego przypadku w osobnych kopiach. Oceń, jak zmieniają jakość i zakres danych, bez nadpisywania wyniku podstawowego.

Walidacja i zapis wyniku

Kontrole przed eksportem

Uporządkuj zbiór w 14 kolumn opisanych poniżej. Uzupełnij docs/slownik_danych.md o nazwy wynikowe, typy, jednostki, definicje pól pochodnych i dopuszczone braki. Nie zastępuj opisu plików źródłowych opisem wyniku; dodaj odrębną sekcję.

Grupa	Kolumny w kolejności wynikowej
Identyfikacja i czas	czas, rok_miesiac, urzadzenie
Kartoteka	typ_urzadzenia, lokalizacja, moc_nominalna_kw, data_instalacji, wiek_urzadzenia_dni
Pomiary	temperatura, drgania, energia, produkcja
Kontekst pracy	tryb, status

- Porównaj liczbę wierszy ze stanem przed połączeniem. Każdą wcześniejszą zmianę liczby rekordów uzgodnij z dziennikiem.
- Sprawdź kompletność i unikalność klucza, metadane, typy, kategorie oraz reguły zakresów. Pola wymagane kontroluj także pod kątem braków.
- Uzupełnij audyt przed i po. Sprawdź, czy zmiany liczby braków oraz sum wartości wynikają wyłącznie z zapisanych decyzji.

Eksport i ponowne wczytanie

Zapisuj tylko wynik, który przeszedł kontrole. SCHEMAT_WYNIK to słownik, który przygotujesz dla wszystkich kolumn poza dwiema datami: teksty jako string lub category, liczby pomiarowe i moc jako float64, a produkcja i wiek jako Int64. CSV nie przechowuje tych typów.

```
PROCESSED_DIR.mkdir(parents=True, exist_ok=True)
plik_wynik = PROCESSED_DIR / "pomiary_clean.csv"
pomiary_clean.to_csv(
    plik_wynik, index=False, encoding="utf-8",
    date_format="%Y-%m-%d %H:%M"
)
wczytany = pd.read_csv(
    plik_wynik, encoding="utf-8", parse_dates=["czas", "data_instalacji"]
).astype(SCHEMAT_WYNIK)
assert wczytany.shape == pomiary_clean.shape
assert wczytany.columns.tolist() == pomiary_clean.columns.tolist()
```

Przygotuj SCHEMAT_WYNIK przed uruchomieniem tego fragmentu i zastosuj go także do ramki przed eksportem. Po ponownym odczycie powtórz kontrole klucza, braków, kategorii, dat i zakresów. Porównaj wartości tabel, uwzględniając niewielką tolerancję zapisu liczb zmiennoprzecinkowych; sam zgodny kształt nie wystarcza.

Oblicz SHA-256 zapisanego pliku i wpisz go do README.md wraz z datą i ścieżką. Skrót identyfikuje konkretną wersję pliku, ale nie dowodzi jakości danych. Przy kolejnej świadomej zmianie wyniku zaktualizuj wpis.

Zapisz notebook, wykonaj Restart Kernel and Run All Cells, sprawdź pierwszy ewentualny błąd i powtórz cały przebieg. Po udanym wykonaniu zapisz notebook ponownie, aby zachować aktualne wyniki.

Rezultaty i samokontrola

Nazewnictwo plików

Zachowaj nazwy źródeł i katalogów z M1. Pliki nazywaj bez spacji, polskich znaków i dopisków final2. Prefiks M2 oznacza etap pracy; pomiary_clean.csv pozostaje stałym wejściem do kolejnych modułów.

Zapisz w projekcie	Zawartość
notebooks/M2_przygotowanie_danych.ipynb	Kod, wyniki kontroli i dziennik decyzji.
data/processed/pomiary_clean.csv	Zweryfikowany zbiór do dalszej analizy.
outputs/tables/M2_audyt_jakosci.csv	Jedna tabela z kontrolami przed i po zmianach.
docs/slownik_danych.md oraz README.md	Kontrakt wyniku, sposób odtworzenia i SHA-256.

Audyt przed i po zapisz bez indeksu w jednym pliku M2_audyt_jakosci.csv. Obowiązuje folder outputs/tables zgodny z M1, także tam, gdzie slajdy pokazują output/tabele.

Użycie LLM

Przekaż modelowi minimalny kontekst bez danych poufnych. Zweryfikuj sugestię na danych lub w dokumentacji i zapisz werdykt w docs/rejestr_uzycia_LLM.csv, z polem modul ustawionym na M2. Brak użycia LLM odnotuj w notebooku.

Przykład polecenia: „Łączę wiele pomiarów z jednym opisem urządzenia po kluczu urzadzenie. Zaproponuj kontrole przed i po left merge. Nie zakładaj wyników liczbowych. Przy każdej kontroli wyjaśnij, jaki błąd wykrywa”.

Lista kontroli przed zakończeniem modułu

- □ Źródła i notebook M1 pozostały bez zmian, a M2 działa niezależnie od ich sesji.
- □ Każda decyzja jakościowa ma dowód, uzasadnienie i kontrolę skutku.
- □ Przykłady kontrolowane pozostają oddzielone od głównego zbioru.
- □ Połączenie spełnia relację m:1, a niedopasowania są wyjaśnione.
- □ Pola pochodne i końcowy schemat są opisane w słowniku danych.
- □ Ponowny odczyt odtwarza wartości, braki i typy zgodnie z kontraktem.
- □ Audyt, README i rejestr LLM są aktualne; pełne uruchomienie M2 działa.

Gdy pojawi się problem

Jedna kolumna po imporcie CSV: sprawdź separator. Brak kolumny urzadzenie w XLSX: sprawdź arkusz. MergeError lub dodatkowe wiersze: sprawdź klucz kartoteki. NaT po konwersji: obejrzyj źródłowe daty. Różnica po eksporcie: sprawdź indeks, odtworzenie typów i precyzję liczb.

Na koniec zapisz: „Moja najważniejsza decyzja to … . Jej wpływ sprawdziłem przez …”.