Rapport de Projet : Pipeline ETL pour Données de Production Manufacturière

MANUFACTURING ETL PIPELINE

Production Line Data Integration

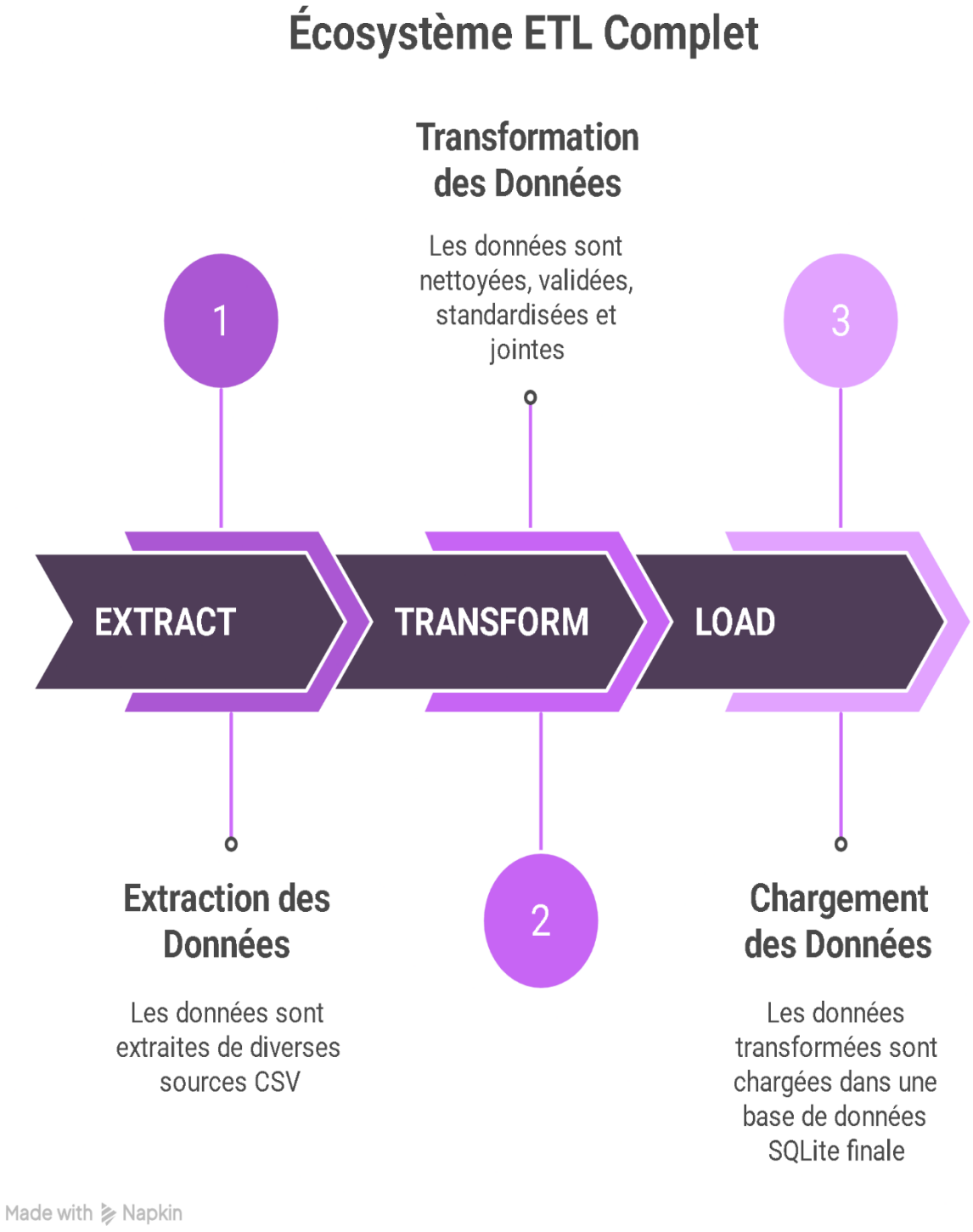
Équipe de Projet :

- **Leader** : Walid FEKIK
  - **Membres** :
    - Oussama ABDELHALIM
    - Abdelmalek BENATTIA

Date de Soumission : Décembre 2025

Institution : ENSTA

Section 1 : Diagramme de Flux de Données :



Section 2 : Logique de Transformation :

2.1 Nettoyage des Données Capteurs :

Règles appliquées :

1. **Valeurs d'erreur → NaN** : -999, -1, NULL, 'null'
2. **Validation plages** :
    - Température : 0°C à 150°C
    - Pression : 0 à 10 bar
    - Vibration : 0 à 100 mm/s
3. **Forward fill** : Valeurs manquantes remplacées par précédente valide
4. **Marquage qualité** :
    - "good" : données valides
    - "estimated" : valeurs estimées (forward fill)
    - "invalid" : valeurs hors plage (rejetées)

Code clé :

```
# Validation des plages
ranges = {
    'temperature': (0, 150),
    'pressure': (0, 10),
    'vibration': (0, 100)
}
```

```
for col, (min_val, max_val) in ranges.items():
    invalid_mask = (df[col] < min_val) | (df[col] > max_val)
    df.loc[invalid_mask, 'data_quality'] = 'invalid'
    df.loc[invalid_mask, col] = np.nan
```

2.2 Standardisation :

Transformations :

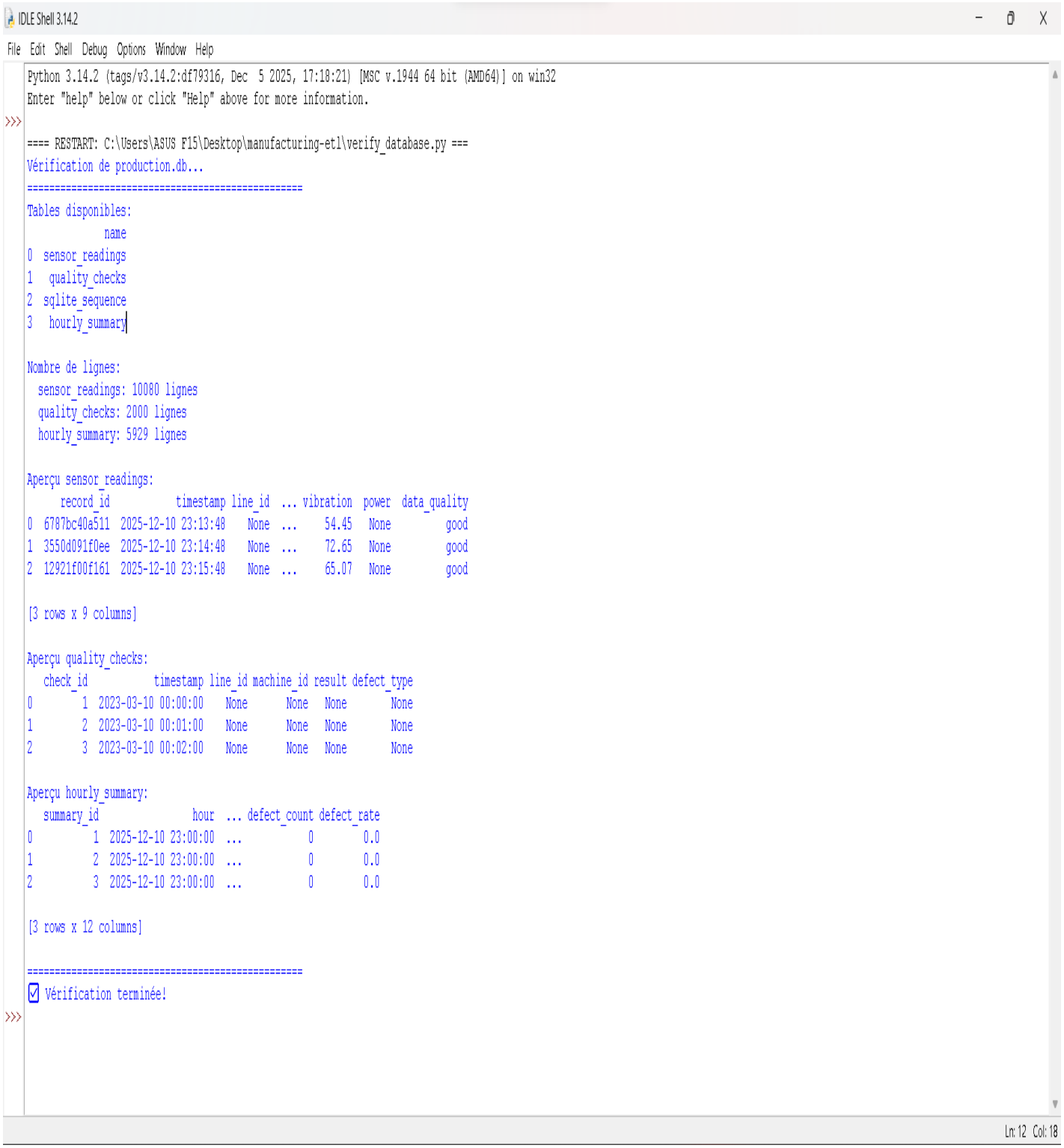
1. **Timestamps** → format datetime pandas
2. **Textes** → minuscules uniformes
3. **Noms colonnes** : espaces → underscores, minuscules
4. **IDs uniques** : génération MD5 basée sur timestamp+machine_id

2.3 Jointure Capteurs-Qualité (Opération Critique)

Stratégie : LEFT JOIN

```
merged_df = pd.merge(
    sensor_df,          # Données capteurs
    quality_df,         # Données qualité
    on=['timestamp', 'machine_id'],  # Clés de jointure
    how='left',         # LEFT JOIN : garde tous capteurs
    suffixes=('', '_quality')
)
```

Section 3: Sample queries and results (screenshots from your database):



• *Section 4: Challenges and solutions:*

Problème identifié :

- Capteurs : timestamps 2025
- Qualité : timestamps 2023
- → 0 correspondances trouvées

Solution implémentée :

1. Gestion NULL : Colonne quality_status avec valeur par défaut
2. Statut par défaut : 'not_checked' quand pas de correspondance
3. Logging : Information claire sur le mismatch

Fonction de détermination du statut :

```
def get_quality_status(result):
    if pd.isna(result):
        return 'not_checked'
    elif 'pass' in str(result).lower():
        return 'passed'
    elif 'fail' in str(result).lower():
        return 'failed'
    else:
        return 'unknown'
```

Résultats jointure :

- Lignes capteurs : 10,080
- Correspondances trouvées : 0 (100% mismatch temporel)
- Lignes avec quality_status='not_checked' : 10,080

2.4 Agrégations Horaires

Groupement : Par heure, machine_id, line_id

Calculs par groupe :

- Statistiques : moyenne, min, max, écart-type
- Température, pression, vibration
- Métriques qualité : total_checks, defect_count, defect_rate

Formule taux défaut :

defect_rate = (defect_count / total_checks) × 100

Si total_checks = 0 → defect_rate = 0%

Résultats agrégation :

- Agrégations calculées : 5,929
- Période : 2025-12-10 à 2025-12-17
- Colonnes générées : 18 (3 IDs + 12 stats + 3 métriques)