Hakim RAHHOU
Rapport du projet Data Engineer

Encadré par M. Kévin letup

2025- 2026

Test sur les deux requettes :

```sql
-- Nb d'emplacements disponibles de vélos dans une ville
SELECT dm.NAME, tmp.SUM_BICYCLE_DOCKS_AVAILABLE
FROM DIM_CITY dm INNER JOIN (
    SELECT CITY_ID, SUM(BICYCLE_DOCKS_AVAILABLE) AS
SUM_BICYCLE_DOCKS_AVAILABLE
    FROM FACT_STATION_STATEMENT
    WHERE CREATED_DATE = (SELECT MAX(CREATED_DATE) FROM CONSOLIDATE_STATION)
    GROUP BY CITY_ID
) tmp ON dm.ID = tmp.CITY_ID
WHERE lower(dm.NAME) in ('paris', 'nantes', 'vincennes', 'toulouse');
```

Results:

```
PS C:\Users\hakim\OneDrive\Desktop\IS5\Data Enigineering\de-projet\Data_Eng_Project> .\duckdb.exe .\data\duckdb\mobility_analysis.duckdb
D SELECT dm.NAME, tmp.SUM_BICYCLE_DOCKS_AVAILABLE
┌   FROM DIM_CITY dm
┌   INNER JOIN (
┌       SELECT CITY_ID, SUM(BICYCLE_DOCKS_AVAILABLE) AS SUM_BICYCLE_DOCKS_AVAILABLE
┌       FROM FACT_STATION_STATEMENT
┌       WHERE CREATED_DATE = (SELECT MAX(CREATED_DATE) FROM CONSOLIDATE_STATION)
┌       GROUP BY CITY_ID
┌   ) tmp
┌   ON dm.ID = tmp.CITY_ID
┌   WHERE lower(dm.NAME) IN ('paris', 'nantes', 'vincennes', 'toulouse');
```

NAME varchar	SUM_BICYCLE_DOCKS_AVAILABLE int128
Vincennes	127
Nantes	1463
Paris	20168

Requête 1 : Somme des bornes disponibles par ville

NAME	SUM_BICYCLE_DOCKS_AVAILABLE
Paris	18890
Vincennes	170
Nantes	1072

```sql
-- Nb de vélos disponibles en moyenne dans chaque station
SELECT ds.name, ds.code, ds.address, tmp.avg_dock_available
FROM DIM_STATION ds JOIN (
    SELECT station_id, AVG(BICYCLE_AVAILABLE) AS avg_dock_available
    FROM FACT_STATION_STATEMENT
    GROUP BY station_id
) AS tmp ON ds.id = tmp.station_id;
```

Results:

```
PS C:\Users\hakim\OneDrive\Desktop\IS5\Data Enigineering\de-projet\Data_Eng_Project> .\duckdb.exe .\data\
duckdb\mobility_analysis.duckdb
──────────────────────────────────────────────────┘D
D SELECT
T     ds.name,
T     ds.code,
T     ds.address,
T     tmp.avg_dock_available
T FROM DIM_STATION ds
T JOIN (
T     SELECT station_id, AVG(BICYCLE_AVAILABLE) AS avg_dock_available
T     FROM FACT_STATION_STATEMENT
T     GROUP BY station_id
T ) AS tmp
T ON ds.id = tmp.station_id;
```

NAME varchar	CODE varchar	ADDRESS varchar	avg_dock_available double
Saint-Sulpice	6003	NULL	11.666666666666666
Place Nelson Mandela	25006	NULL	2.6666666666666665
Jules Guesde - Pont du Port à l'Anglais	44017	NULL	15.333333333333334
Gare de Nogent-le-Perreux	41303	NULL	21.333333333333332
Boulangers - Cardinal Lemoine	5022	NULL	8.666666666666666
Commandant Schloesing - Pétrarque	16202	NULL	0.0
Quai de la Seine	19003	NULL	9.666666666666666
Sibelle - Alésia	14012	NULL	6.333333333333333
Place de Barcelone - Mirabeau	16030	NULL	44.333333333333336
Général De Gaulle - Alouette	41601	NULL	9.333333333333334
Marseille - Beaurepaire	10014	NULL	26.666666666666668
Auguste Cain - Jean Moulin	14136	NULL	4.333333333333333
Félix Faure - Sadi Carnot	33009	NULL	15.333333333333334
Place Violet	15033	NULL	14.333333333333334
18 juin 1940 - Buzenval	25005	NULL	11.0
Anatole France - Jean Lolive	35019	NULL	8.666666666666666
Gravelle - Route du Bac	12126	NULL	11.666666666666666
Pascal - Claude Bernard	5026	NULL	27.333333333333332
Labouret - Saint-Denis	27008	NULL	16.333333333333332
Capitaine Glarner - Gabriel Péri	34002	NULL	13.0
.	.	.	.
.	.	.	.
.	.	.	.

```
PS C:\Users\hakim\OneDrive\Desktop\IS5\Data Enigineering\de-projet\Data_Eng_Project> .\duckdb.exe .\data\
duckdb\mobility_analysis.duckdb
```

Sibelle - Alésia	14012	NULL	6.333333333333333
Place de Barcelone - Mirabeau	16030	NULL	44.333333333333336
Général De Gaulle - Alouette	41601	NULL	9.333333333333334
Marseille - Beaurepaire	10014	NULL	26.666666666666668
Auguste Cain - Jean Moulin	14136	NULL	4.333333333333333
Félix Faure - Sadi Carnot	33009	NULL	15.333333333333334
Place Violet	15033	NULL	14.333333333333334
18 juin 1940 - Buzenval	25005	NULL	11.0
Anatole France - Jean Lolive	35019	NULL	8.666666666666666
Gravelle - Route du Bac	12126	NULL	11.666666666666666
Pascal - Claude Bernard	5026	NULL	27.333333333333332
Labouret - Saint-Denis	27008	NULL	16.333333333333332
Capitaine Glarner - Gabriel Péri	34002	NULL	13.0
.	.	.	.
.	.	.	.
.	.	.	.
084-RÉGION	84	Rue de la Loire - place Gaston Defferre	8.0
060-GARE DE NANTES NORD	60	27 boulevard Stalingrad	30.0
053-OLIVETTES	53	10, rue des Olivettes	1.0
086-HAUTS PAVÉS	86	Rue de Berry - Rue des Hauts Pavés	13.0
093-FACULTÉS	93	A l'angle route de la Jonelière - Boulevard Guy Mollet	8.0
106-GARENNES	106	2, rue des Garennes	11.0
097-SAINT DONATIEN	97	Place du 51ème Régiment d'Artillerie	12.0
031-BOURSE	31	Allée de la Bourse - Boulevard des Nations-Unies	4.0
041-BRUNELIERE	41	88, quai de la Fosse	6.0
071-TALENSAC NORD	71	18, rue Talensac - 26, rue de Bel Air	10.0
066-CHANZY	66	Face au 23, avenue Chanzy	3.0
112-TRENTEMOULT SABLIERES	112	Rue de la Californie, 44400 REZE	6.0
109-CLOS TOREAU	109	Face 31, boulevard Emile Gabory	7.0
040-MADELEINE	40	Quai Moncousu - Pont Général Audibert	6.0
026-GUIST'HAU SUD	26	30, boulevard Gabriel Guist'Hau - Rue Harouys	10.0
019-SAINT SIMILIEN	19	1, place Saint Similien	4.0
057-GAETAN RONDEAU	57	20, rue Gaëtan Rondeau	9.0
119-ZOLA	119	Place Emile Zola	22.0
070-GARE DE NANTES SUD	70	Quai de Malakoff - Canal Saint-Félix	45.0
111-DIDEROT	111	Avenue de la Vendée - Face à l'Espace Diderot - REZÉ	11.0

1632 rows (40 shown) 4 columns

```
D
```

Récupérer **le nombre d'habitants d'une ville :**

```
SELECT * FROM CONSOLIDATE_CITY;
```

```
PS C:\Users\hakim\OneDrive\Desktop\IS5\Data Enigineering\de-projet\Data_Eng_Pr
DuckDB v1.4.1 (Andium) b390a7c376
Enter ".help" for usage hints.
D SELECT * FROM CONSOLIDATE_CITY;
```

ID varchar	NAME varchar	NB_INHABITANTS int32	CREATED_DATE varchar
93066	Saint-Denis	115237	2025-12-05
94041	Ivry-sur-Seine	64526	2025-12-05
94018	Charenton-le-Pont	28756	2025-12-05
92007	Bagneux	43647	2025-12-05
94081	Vitry-sur-Seine	95282	2025-12-05
93006	Bagnolet	41776	2025-12-05
94076	Villejuif	58142	2025-12-05
94017	Champigny-sur-Marne	78367	2025-12-05
93061	Le Pré-Saint-Gervais	16733	2025-12-05
94043	Le Kremlin-Bicêtre	23678	2025-12-05
92026	Courbevoie	81945	2025-12-05
94002	Alfortville	45569	2025-12-05
94042	Joinville-le-Pont	20784	2025-12-05
92033	Garches	17705	2025-12-05
94073	Thiais	32006	2025-12-05
92009	Bois-Colombes	29376	2025-12-05
94054	Orly	24488	2025-12-05
93039	L'Ile-Saint-Denis	8682	2025-12-05
93063	Romainville	35314	2025-12-05
92049	Montrouge	46273	2025-12-05
·	·	·	·
·	·	·	·
·	·	·	·
92073	Suresnes	48932	2025-12-05
94069	Saint-Maurice	14411	2025-12-05
92023	Clamart	56882	2025-12-05
92020	Châtillon	36224	2025-12-05
92077	Ville-d'Avray	10871	2025-12-05
94028	Créteil	92859	2025-12-05
92063	Rueil-Malmaison	80842	2025-12-05
93048	Montreuil	110758	2025-12-05
94080	Vincennes	48368	2025-12-05

🚲 Pipeline ETL – Analyse des stations de vélos (Paris + Nantes)

📌 Description du projet

Ce projet implémente un **pipeline ETL complet** permettant :

- L'ingestion de données open-data en temps réel
- La consolidation dans une base **DuckDB**
- La modélisation **dimensionnelle (Data Warehouse)**
- L'analyse des stations de vélos en libre-service

Le pipeline était initialement fourni pour **Paris**, et j'ai enrichi le projet avec :
✓ Les données temps réel de la ville de Nantes
✓ L'intégration multi-ville dans les tables de consolidation
✓ Les dimensions et faits pour Paris + Nantes
✓ Les requêtes analytiques demandées dans le sujet

🏗️ Architecture du projet

```
Data_Eng_Project/
│
├── src/
│   ├── data_ingestion.py
│   ├── data_consolidation.py
│   ├── data_agregation.py
│   ├── main.py
│
├── data/
│   ├── raw_data/
│   │   └── YYYY-MM-DD/
│   │       ├── paris_realtime_bicycle_data.json
│   │       ├── nantes_realtime_bicycle_data.json
│   ├── duckdb/
│       └── mobility_analysis.duckdb
│
├── data/sql_statements/
│   ├── create_consolidate_tables.sql
│   ├── create_agregate_tables.sql
│
└── README.md
```

⚙ Fonctionnement du pipeline

1 Ingestion

📄 Fichier : `data_ingestion.py`

- Récupère les données Velib Paris via API open-data
- Récupère les données Nantes via l'endpoint Naolib
- Stocke les données dans `/data/raw_data/YYYY-MM-DD/`

```
get_paris_realtime_bicycle_data()
get_nantes_realtime_bicycle_data()
```

---

2 Consolidation

📄 Fichier : `data_consolidation.py`

Création des tables :

- `CONSOLIDATE_CITY`
- `CONSOLIDATE_STATION`
- `CONSOLIDATE_STATION_STATEMENT`

Intégration multi-villes :
√ Paris
√ Nantes

Chaque station est enrichie :

- coordonnées GPS
- capacité
- statut
- code INSEE de la ville
- date d'ingestion

---

3 Agrégation (modèle dimensionnel)

📄 Fichier : `data_agregation.py`

Tables créées :

📌 Dimensions

- `DIM_CITY`
- `DIM_STATION`

📌 Fait

- `FACT_STATION_STATEMENT`
  (une ligne = disponibilité d'une station un jour donné)

▶ Exécution du pipeline

```
git clone <repo>
cd Data_Eng_Project

python -m venv .venv
source .venv/bin/activate  # Linux/Mac
# OU
.\.venv\Scripts\activate    # Windows

pip install -r requirements.txt

python src/main.py
```

📊 Requêtes finales (validations)

📌 1. Nombre total de places disponibles par ville

```
SELECT dm.NAME, tmp.SUM_BICYCLE_DOCKS_AVAILABLE
FROM DIM_CITY dm INNER JOIN (
    SELECT CITY_ID, SUM(BICYCLE_DOCKS_AVAILABLE) AS
SUM_BICYCLE_DOCKS_AVAILABLE
    FROM FACT_STATION_STATEMENT
    WHERE CREATED_DATE = (SELECT MAX(CREATED_DATE) FROM CONSOLIDATE_STATION)
    GROUP BY CITY_ID
) tmp ON dm.ID = tmp.CITY_ID
WHERE lower(dm.NAME) in ('paris', 'nantes', 'vincennes');
```

✓ Résultats obtenus

Ville	Places disponibles
Paris	18890
Nantes	1072
Vincennes	170

📌 2. Moyenne de vélos disponibles par station

```
SELECT ds.name, ds.code, ds.address, tmp.avg_dock_available
FROM DIM_STATION ds JOIN (
    SELECT station_id, AVG(BICYCLE_AVAILABLE) AS avg_dock_available
    FROM FACT_STATION_STATEMENT
    GROUP BY station_id
) AS tmp ON ds.id = tmp.station_id;
```

📡 Schéma du pipeline