

Projet SQL B2

Système de gestion de locations de véhicules
électriques partagés



clAra Mobility : Notre contexte

clAra Mobility est une start-up innovante spécialisée dans la **mobilité urbaine durable**. Elle propose un service moderne de location de véhicules électriques partagés incluant voitures, trottinettes, scooters et vélos électriques.

Face à une croissance rapide, l'entreprise doit centraliser la gestion de sa flotte, ses utilisateurs, les bornes de recharge, les réservations, paiements et opérations de maintenance.



Votre mission : créer une base de données complète

01

Analyse et modélisation

Concevoir le système selon la méthodologie Merise avec MCD et MLD

02

Création de la base

Implémenter la structure PostgreSQL avec contraintes d'intégrité

03

Requêtes SQL

Développer des requêtes avancées, vues et triggers

04

Gestion de versions

Collaborer via GitHub avec commits réguliers

05

Documentation

Produire une documentation claire et professionnelle



Pourquoi ce projet est crucial pour vous

Compétences techniques

Maîtrisez PostgreSQL, SGBD très demandé en 2025-2026 pour les postes de data analyst, data engineer et développeur backend

Cas d'usage moderne

Plongez dans un contexte réel : mobilité électrique, IoT et gestion de données à grande échelle

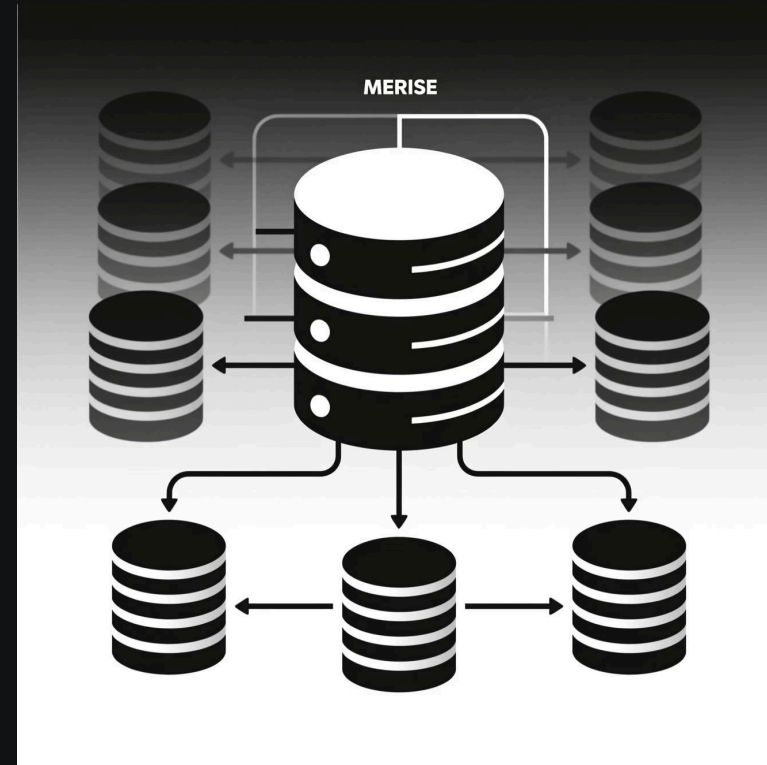
Travail collaboratif

Développez vos compétences en gestion de versions, communication et documentation professionnelle

Mission 1 : Analyse et modélisation Merise

Entités à identifier

- Véhicules électriques (types, statuts, caractéristiques)
- Stations et bornes de recharge
- Clients et techniciens
- Locations et réservations
- Paiements et transactions
- Interventions de maintenance



📄 **Livrable attendu :** Document PDF contenant le MCD, MLD et dictionnaire de données complet

Mission 2 : Création de la base PostgreSQL



Créer la structure

Implémenter les tables selon le MLD validé



Contraintes d'intégrité

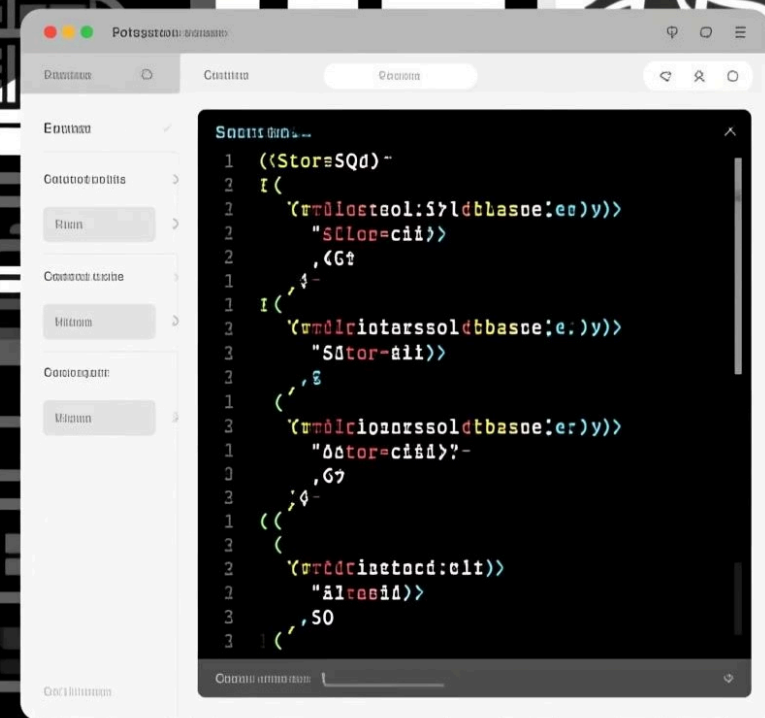
PRIMARY KEY, FOREIGN KEY, CHECK, NOT NULL



Import des données

Charger le CSV fourni et créer des données test

Livrable attendu : Script SQL create_tables.sql + données importées et testées



Mission 3 : Maîtriser les requêtes SQL avancées

1

Requêtes de base

SELECT avec WHERE, JOIN pour combiner les tables

2

Agrégations

COUNT, AVG, MAX pour analyser les données

3

Sous-requêtes

Requêtes imbriquées complexes

4

Vues

Créer des vues pour simplifier l'accès aux données

5

Triggers

Automatiser la mise à jour du statut des véhicules

6

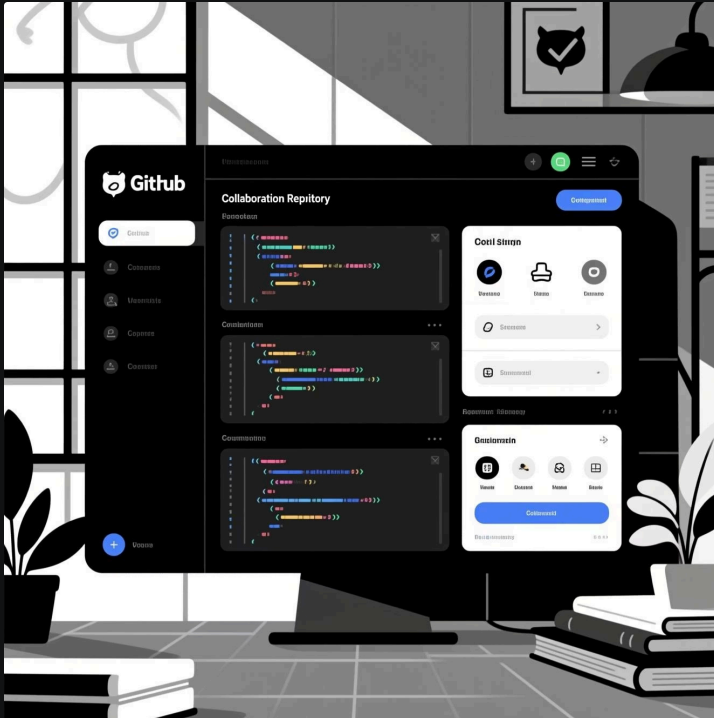
Fonctions

Calculer le coût total selon durée et type

Minimum 10 requêtes démontrant votre maîtrise de SQL. Livrable : fichier queries.sql documenté

Mission 4 & 5 : GitHub et Documentation

Gestion de versions



- Créer un dépôt GitHub partagé
- Commits réguliers avec messages clairs
- Historique visible de la collaboration

Documentation professionnelle



- Rédiger un README.md complet
- Expliquer le contexte et le schéma
- Inclure les instructions de lancement

PostgreSQL : L'outil professionnel par excellence



Open source et gratuit

Parfait pour l'apprentissage et accessible à tous, sans coût de licence



Performant et robuste

Utilisé par Spotify, Reddit et de nombreuses start-ups innovantes



Standard SQL

Compétences transférables vers d'autres SGBD du marché



Fonctionnalités avancées

JSONB, géolocalisation, vues matérialisées, triggers pour la pédagogie

Modalités de rendu et présentation



Date limite

Mercredi 7 janvier 2026 à 23h59

Pénalité : -2 points par jour de retard



Présentation orale - 10 min

Jeudi 8 janvier 2026 dès 8h45

5 min présentation + 5 min questions



Livrables Moodle

Lien GitHub, noms des membres, date du dernier push

Préparez un support de présentation professionnel (slides ou démonstration live) pour valoriser votre travail !



Grille d'évaluation

Critère	Détail	Points
A. Partie technique (10 pts)		
Modélisation Merise	MCD + MLD complets et cohérents	2
Création du schéma SQL (PostgreSQL)	Contraintes, clés, structure	2
Qualité des requêtes SQL	Complexité, précision, pertinence	3
Gestion de versions Git	Activité, rigueur, commits réguliers	2
Présentation physique et propreté du rendu	README, clarté, rigueur du dépôt	1
B. Partie orale (10 pts)		
Clarté et structuration du discours		2
Maîtrise technique (SQL, Merise, PostgreSQL)		3
Répartition de la parole / travail d'équipe		2
Capacité à répondre aux questions		2
Présentation physique (attitude, tenue, aisance)		1
TOTAL		/20