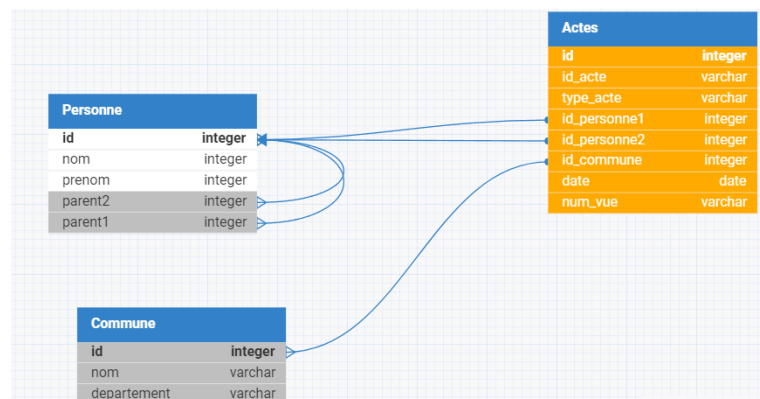


Projet - Modélisation de base de données

Explication :

Le programme est composé de trois parties : la récupération des données CSV, le traitement (que l'on verra en détail plus tard) et la restitution des données dans un fichier SQL nommé "Output.sql".

Le principal but est de pouvoir importer le fichier CSV et de pouvoir travailler dessus, nous avons donc commencé par faire un plan de la base de donnée à utiliser. Voici donc le modèle relationnel sur lequel nous travaillerons.



Nous commençons donc par récupérer les données du csv puis par les mettre dans des caches avant de les traiter. Nous les traitons ensuite grâce à des requêtes généralistes auxquelles on ajoute leurs valeurs ensuite.

Problème n°1 :

Il existe des parents et des enfants avec le même prénom ...

Solution :

Implémenter les personnes en ayant trié la liste par date auparavant. (Les parents seront toujours avant leur enfants).

Requêtes des généalogistes :

1. La quantité de communes par département

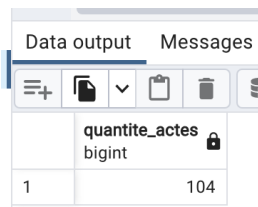
```
SELECT departement, COUNT(*) AS quantite_communes FROM Commune GROUP BY departement;
```



	departement character varying	quantite_communes bigint
1	85	313
2	49	2
3	44	9
4	79	51

2. La quantité d'actes à "LUÇON"

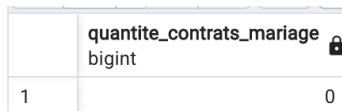
```
SELECT COUNT(*) AS quantite_actes FROM Acte JOIN Commune ON Acte.commune_id = Commune.id WHERE Commune.nom = 'LUÇON';
```



	quantite_actes bigint
1	104

3. La quantité de "contrats de mariage" avant 1855

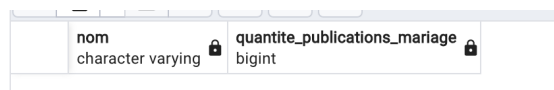
```
SELECT COUNT(*) AS quantite_contrats_mariage FROM Acte WHERE type_acte = 'contrat de mariage' AND date < '01-01-1855';
```



	quantite_contrats_mariage bigint
1	0

4. La commune avec la plus quantité de "publications de mariage"

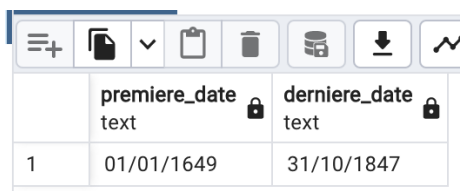
```
SELECT Commune.nom, COUNT(*) AS quantite_publications_mariage FROM Acte JOIN Commune ON Acte.commune_id = Commune.id WHERE type_acte = 'publication de mariage' GROUP BY Commune.nom ORDER BY COUNT(*) DESC LIMIT 1;
```



	nom character varying	quantite_publications_mariage bigint
--	--------------------------	---

5. La date du premier acte et le dernier acte

```
SELECT MIN(date) AS premiere_date, MAX(date) AS derniere_date FROM Acte;
```



	premiere_date text	derniere_date text
1	01/01/1649	31/10/1847

Annexe :

~~Id~~
~~Type~~
~~Nom personne 1~~

...	3	4	15	...
	Nom Personne A	Prénom Personne A		Date	

Donc with B (MOM Personne B : 8
Prénom Personne B : 9)

Personne A
Nom = ~~Mc~~ Colone (3)
Prénom = Colone (4)

Date (Colone 15) ± 25 ans
Parent (P)
Parent 2 (P)

Personne B
Nom = Colone (8)
Prénom = Colone (9)

↳ Vérifications par parents,
~~IMPOSSIBLE~~

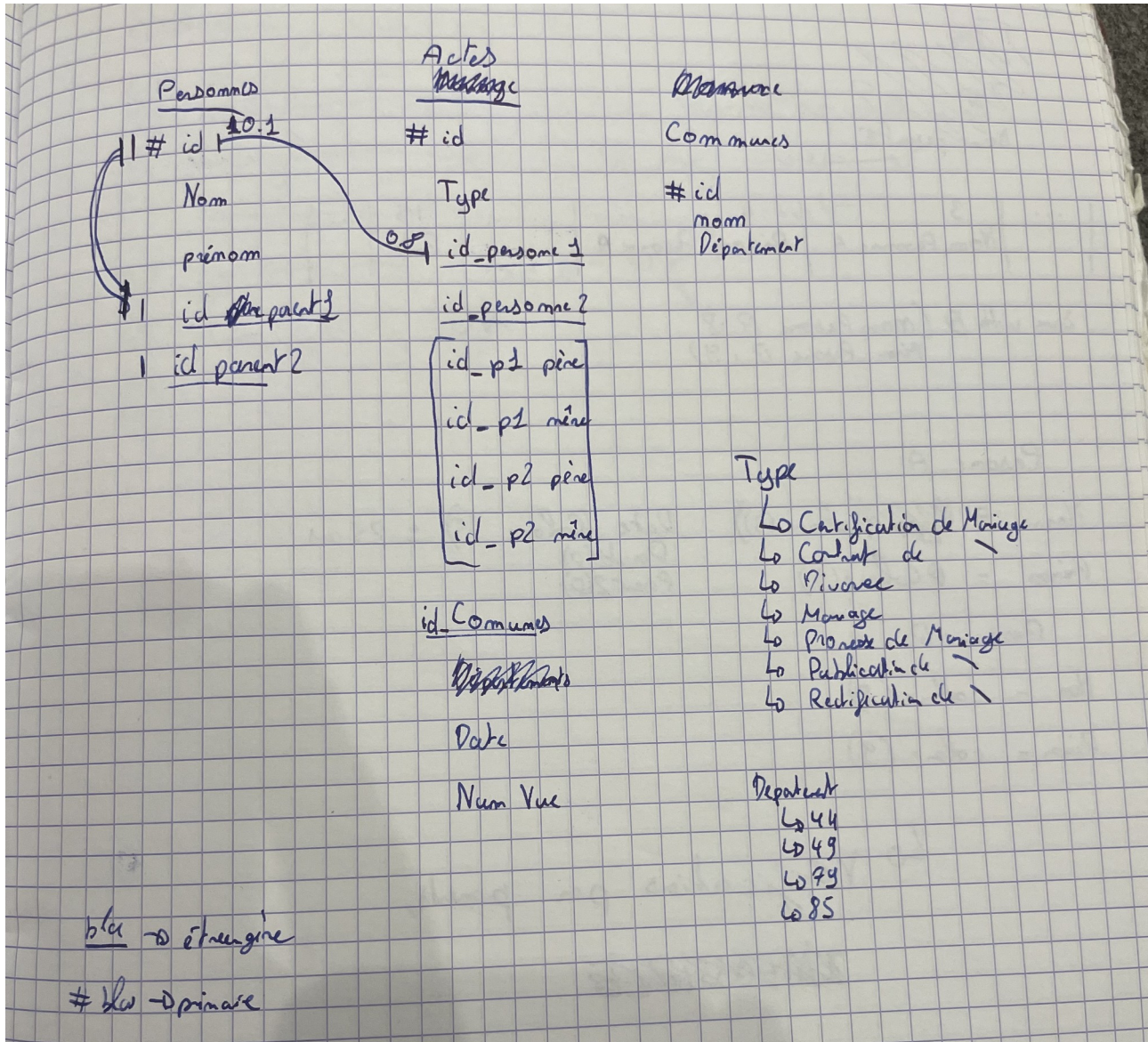
-1
sur les

Personne A)

- ↳ MOM : Colone (3)
- ↳ Prénom : Colone (4)
- ↳ P2 -> Parent Nom 1
- > Parent Prénom 1 : Colone (5)
- ↳ P2 -> Parent Nom 2 : Colone (6)
- > Parent Prénom 2 : Colone (7)

Personne B)

- ↳ Nom : Colone (8)
- ↳ Prénom : Colone (9)
- ↳ P1 -> Parent 1 Nom
- > Parent 1 Prénom : (10)
- ↳ P2 -> Parent 2 Nom (11)
- > Parent 2 Prénom



~~Table~~
Type
Nom persone A

...	3	4	15	...
	Nom Personne A	Prénom Personne A		Date	

Donc with B (nom Personne B : 8
Prénom Personne B : 9)

Personne A

Nom = ~~colonne (3)~~

Prénom = colonne (4)

Date (colonne 15) ± 25 ans
Parent (P)
Parent 2 (O)

Personne B

Nom = colonne (8)

Prénom = colonne (9)

↳ Vérifications par parents,

~~IMPOSSIBLE~~

-1
swaps

Personne A)

↳ NOM : colonne (3)

↳ Prénom : colonne (4)

↳ P₂ → Parent Nom 1

→ Parent Prénom 1 : colonne (5)

↳ P₂ → Parent Nom 2 : colonne (6)

→ Parent Prénom 2 : colonne (7)

Personne B)

↳ Nom : colonne (8)

↳ Prénom : colonne (9)

↳ P₁ → Parent 1 Nom

→ Parent 1 Prénom : (10)

↳ P₂ → Parent 2 Nom (11)

→ Parent 2 Prénom

VANTARD Timothé
WENDLING Valentin