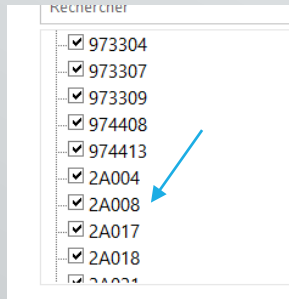


Requêter une base de données SQL

De la création aux résultats

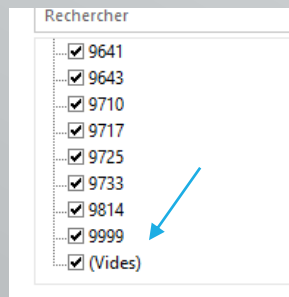
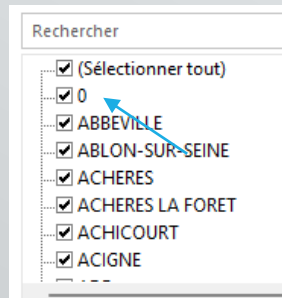
La compréhension des données



Pour les codes communes, il y a finalement quelques **données textes**

- Pour chaque colonne des différents tableau, déterminer la **nature des données** et comprendre leur intérêt.
- Filtrer les colonnes pour détecter **les pièges**.
 - Exemples :

Pour les communes il y a des **données d'erreurs**, rédigées sous la forme d'un « 0 »



Pour les numéros de voies, il y a des **données vides**

- Caractères textes cachés
- Valeurs erronées
- Valeurs vides

La création du dictionnaire

	Nom des colonnes	Type de donnée	Taille	Clé	Description
CONTRAT.CSV	Contrat_ID	INTEGER		Clé primaire	Id unique pour les contrats
	No_voie	CHAR	4		Numéro dans la voie pour l'adresse du logement assuré
	B_T_Q	CHAR	1		Indicateur éventuel de répétition pour l'adresse du logement assuré sur un caractère
	Type_de_voie	CHAR	4		Type de voie pour l'adresse du logement assuré: rue, av (Avenue), rte (Route), ...
	Voie	VARCHAR			Libellé de la voie pour l'adresse du logement assuré
	Code_dep_code_commune	CHAR	6	Clé étrangère	Concaténation du code département et code commune pour avoir une clé unique
	Code_postal	INTEGER			Code postal pour l'adresse du logement assuré
	Commune	VARCHAR			Libellé de la commune de l'adresse du logement
	Code_département	INTEGER			Numéro du département
	Surface	INTEGER			Surface du bien immobilier
	Type_local	CHAR	11		Typologie du local (appartement, maison)
	Occupation	CHAR	12		Qualité de l'occupant (locataire, propriétaire)
	Type_contrat	CHAR	20		Le type de contrat (Mise en location, Residence principale ou secondaire)
	Formule	CHAR	9		Le type de formule (Intégrale ou classique)
	Valeur_declaree_biens	CHAR	12		La fourchette de valeur du bien
REGION.CSV	Prix_cotisation_mensuel	INTEGER			Montant de la cotisation mensuelle
	Code_dep_code_commune	CHAR	6	Clé primaire	Concaténation du code département et code commune pour avoir une clé unique
	reg_code	INTEGER			Code de la région
	reg_nom	CHAR	26		Nom de la région
	aca_nom	CHAR	24		Nom de l'académie
	dep_nom	CHAR	43		Nom du département
	com_nom_maj_court	VARCHAR			Nom de la commune
	dep_code	CHAR	3		Numéro du département
	dep_nom_num	CHAR	49		Concaténation du nom du département et du numéro du département entre parenthèses

- Compléter le dictionnaire avec tous les **noms de colonnes**
- Reporter le **type** de données et la **description**, collectés lors de l'étape précédente
- Déterminer la **clé primaire** et la **clé étrangère** de chaque table

La création de l'architecture

- A l'aide du dictionnaire crée à l'étape précédente, rédiger **l'architecture** de la base de données sur SQL Power Architect.
- Intégrer **la jointure** entre la clé primaire à la clé étrangère
- Générer le **code SQL** de l'architecture

Contrat
Contrat_ID: INTEGER NOT NULL [PK]
No_Voie: CHAR(4)
B_T_Q: CHAR(1)
Type_de_voie: CHAR(4) NOT NULL
Voie: VARCHAR NOT NULL
Code_dep_code_commune: CHAR(6) NOT NULL [FK]
Code_postal: INTEGER NOT NULL
Commune: VARCHAR NOT NULL
Code_departement: INTEGER NOT NULL
Surface: INTEGER NOT NULL
Type_Local: CHAR(11) NOT NULL
Occupation: CHAR(12) NOT NULL
Type_Contrat: CHAR(20) NOT NULL
Formule: CHAR(9) NOT NULL
Valeur_declaree_biens: CHAR(12) NOT NULL
Prix_cotisation_mensuel: INTEGER NOT NULL

Region
Code_dep_code_commune: CHAR(6) NOT NULL [PK]
reg_Code: INTEGER NOT NULL
reg_nom: CHAR(26) NOT NULL
aca_nom: CHAR(24) NOT NULL
dep_nom: CHAR(43) NOT NULL
com_nom_maj_court: VARCHAR NOT NULL
dep_code: CHAR(3) NOT NULL
dep_nom_num: CHAR(49) NOT NULL

Forward Engineer SQL Script

Create in: PlayPen Database Properties...

Generate DDL for Database Type: PostgreSQL (no catalog)

Schema: public

☐ Generate Liquibase XML

OK Cancel

Preview SQL Script

Your Target Database is not configured.

```
CREATE TABLE public.Region (  
  Code_dep_code_commune CHAR(6) NOT NULL,  
  reg_Code INTEGER NOT NULL,  
  reg_nom CHAR(26) NOT NULL,  
  aca_nom CHAR(24) NOT NULL,  
  dep_nom CHAR(43) NOT NULL,  
  com_nom_maj_court VARCHAR NOT NULL,  
  dep_code CHAR(3) NOT NULL,  
  dep_nom_num CHAR(49) NOT NULL,  
  CONSTRAINT region_pk PRIMARY KEY  
  (Code_dep_code_commune)  
);  
  
CREATE TABLE public.Contrat (  
  Contrat_ID INTEGER NOT NULL,  
  No_Voie CHAR(4),  
  B_T_Q CHAR(1),  
  Type_de_voie CHAR(4) NOT NULL,  
  Voie VARCHAR NOT NULL,  
  Code_dep_code_commune CHAR(6) NOT NULL,  
  Code_postal INTEGER NOT NULL,  
  Commune VARCHAR NOT NULL,  
  Code_departement INTEGER NOT NULL,  
  Surface INTEGER NOT NULL,  
  Type_Local CHAR(11) NOT NULL,  
  Occupation CHAR(12) NOT NULL,  
  Type_Contrat CHAR(20) NOT NULL,  
  Formule CHAR(9) NOT NULL,  
  Valeur_declaree_biens CHAR(12) NOT NULL,  
  Prix_cotisation_mensuel INTEGER NOT NULL,  
  CONSTRAINT contrat_pk PRIMARY KEY (Contrat_ID)  
);  
  
ALTER TABLE public.Contrat ADD CONSTRAINT region_contrat_fk  
FOREIGN KEY (Code_dep_code_commune)  
REFERENCES public.Region (Code_dep_code_commune)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION  
NOT DEFERRABLE;
```

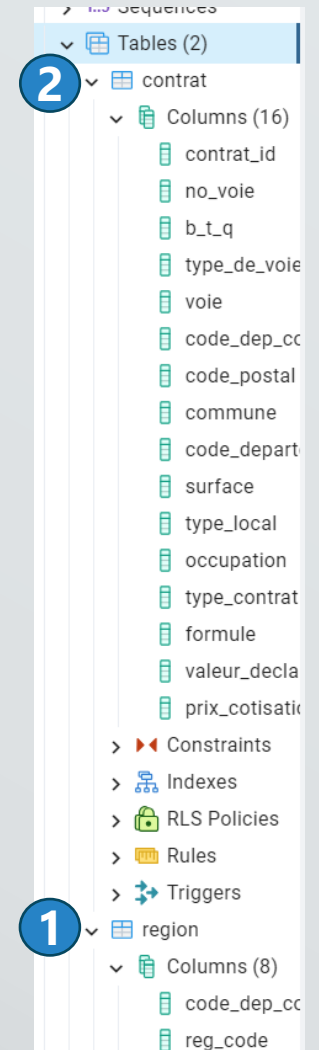
Copy Execute Save Close

La création des tables

Query Query History

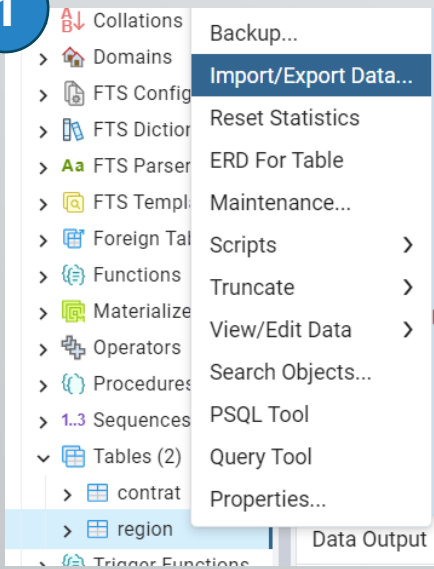
```
1 CREATE TABLE public.Region (  
2     Code_dep_code_commune CHAR(6) NOT NULL,  
3     reg_Code INTEGER NOT NULL,  
4     reg_nom CHAR(26) NOT NULL,  
5     aca_nom CHAR(24) NOT NULL,  
6     dep_nom CHAR(43) NOT NULL,  
7     com_nom_maj_court VARCHAR NOT NULL,  
8     dep_code CHAR(3) NOT NULL,  
9     dep_nom_num CHAR(49) NOT NULL,  
10    CONSTRAINT region_pk PRIMARY KEY (Code_dep_code_commune)  
11 );  
12  
13  
14 CREATE TABLE public.Contrat (  
15     Contrat_ID INTEGER NOT NULL,  
16     No_Voie CHAR(4),  
17     B_T_Q CHAR(1),  
18     Type_de_voie CHAR(4) NOT NULL,  
19     Voie VARCHAR NOT NULL,  
20     Code_dep_code_commune CHAR(6) NOT NULL,  
21     Code_postal INTEGER NOT NULL,  
22     Commune VARCHAR NOT NULL,  
23     Code_departement INTEGER NOT NULL,
```

- Exécuter le Code SQL précédemment généré.
- Nous retrouvons les **deux tables** dans l'architecture du logiciel de SGBD
- A cette étape, il y a les tables, mais **aucune donnée**

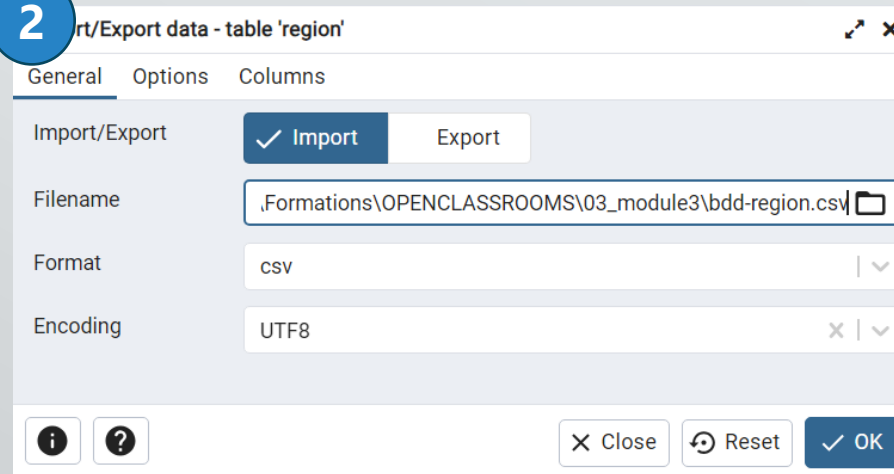


L'import des données

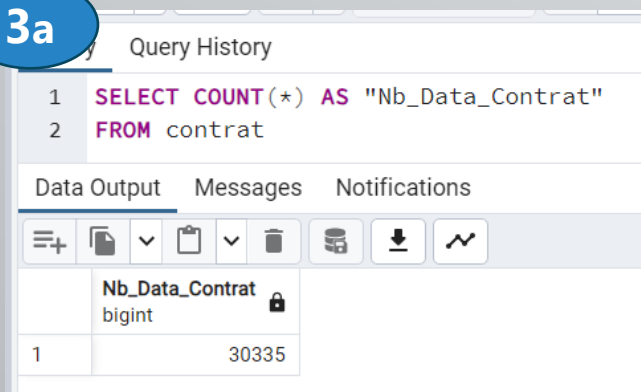
1



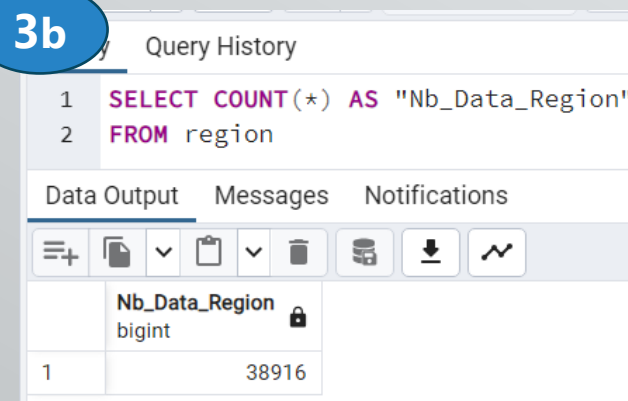
2



3a



3b



- Pour chaque table, **importer** les données en choisissant « Import/Export data »
- Sélectionner **le fichier**, préciser sa **nature** et **l'encodage**.
- **Vérifier l'import** correct des données en faisant la requête « *SELECT COUNT(*) FROM table* »

Les requêtes simples

```
SELECT contrat_id,  
type_contrat, formule  
FROM contrat  
WHERE type_local = 'Maison'  
AND Code_departement = 71
```

	contrat_id [PK] integer	type_contrat character varying	formule character varying
1	114768	Residence principale	Integral
2	114782	Residence principale	Classique
3	114812	Residence principale	Integral
4	114779	Residence principale	Classique

```
SELECT contrat_ID, surface  
FROM contrat  
ORDER BY surface DESC  
LIMIT 5
```

	contrat_id [PK] integer	surface integer
1	104211	815
2	105463	742
3	130878	595
4	100822	570
5	109872	559

```
SELECT DISTINCT reg_nom  
FROM region
```

	reg_nom character varying
1	Hauts-de-France
2	La Réunion
3	Bretagne
4	Mayotte
5	Auvergne-Rhône-Alpes
6	Ile-de-France
7	Grand Est
8	Corse
9	Collectivités d'outre-mer
10	Guyane
11	Bourgogne-Franche-Comté
12	Provence-Alpes-Côte d'Azur
13	Martinique
14	Guadeloupe
15	Normandie
16	Nouvelle-Aquitaine
17	Occitanie
18	Pays de la Loire
19	Centre-Val de Loire

Les requêtes d'agrégation

SELECT
ROUND(AVG(surface),2) AS
"Surface_Moyenne"

FROM contrat

WHERE code_departement =
75

	Surface_Moyenne numeric 🔒
1	51.77

SELECT
ROUND(AVG(Prix_cotisation_m
ensuel),2) AS
"Cotisation_moyenne"

FROM contrat

	Cotisation_moyenne numeric 🔒
1	19.33

SELECT Valeur_declaree_biens,
COUNT(contrat_ID) AS
"Nb_Contrats"

FROM contrat

GROUP BY
Valeur_declaree_biens



	valeur_declaree_biens character varying 🔒	Nb_Contrats bigint 🔒
1	50000-100000	696
2	100000+	104
3	25000-50000	6815
4	0-25000	22720

Les requêtes d'agrégation

```
SELECT count(contrat_ID) AS "Nb_Contrats"  
FROM contrat  
WHERE Type_contrat = 'Residence principale'
```

	Nb_Contrats bigint 
1	25620

```
SELECT Code_departement,  
ROUND(AVG(prix_cotisation_mensuel),2) AS  
"prix_moyen"  
FROM contrat  
GROUP BY Code_departement  
ORDER BY prix_moyen DESC  
LIMIT 10
```

	code_departement integer 	prix_moyen numeric 
1	75	36.40
2	92	26.27
3	94	19.82
4	78	18.88
5	69	18.46
6	1	18.24
7	6	18.14
8	17	17.32
9	74	17.16
10	20	17.03

Les requêtes avec jointures

```
SELECT contrat_id, surface  
FROM contrat ct  
LEFT JOIN region rg ON  
ct.Code_dep_code_commune =  
rg.Code_dep_code_commune  
WHERE com_nom_maj_court = 'CAEN'
```

	contrat_id [PK] integer	surface integer
1	103791	35
2	103792	99
3	103793	40
4	103794	20

```
SELECT COUNT(contrat_ID) AS "Nb_Contrats"  
FROM contrat ct  
LEFT JOIN region rg ON  
ct.Code_dep_code_commune =  
rg.Code_dep_code_commune  
WHERE formule = 'Integral'  
AND reg_nom = 'Pays de la Loire'
```

	Nb_Contrats bigint
1	561

Les requêtes avec jointures

```
SELECT com_nom_maj_court AS  
"Nom_commune",  
COUNT(contrat_ID) AS  
"Nb_Contrats"  
  
FROM contrat ct  
  
LEFT JOIN region rg ON  
ct.Code_dep_code_commune =  
rg.Code_dep_code_commune  
  
GROUP BY com_nom_maj_court  
  
HAVING COUNT(contrat_ID) >= 150  
  
ORDER BY "Nb_Contrats" DESC
```

Nom_commune character varying	Nb_Contrats bigint
PARIS 18	515
PARIS 17	468
PARIS 15	407
PARIS 16	394
NICE	387
PARIS 11	381
BORDEAUX	302
PARIS 20	302
NANTES	291
PARIS 19	266
PARIS 10	263
PARIS 12	252
PARIS 14	222
GRENOBLE	220
PARIS 9	204
TOULOUSE	187
TOULON	170
COURBEVOIE	163
LILLE	161
PARIS 3	159

```
SELECT reg_nom,  
COUNT(contrat_ID) AS  
"Nb_Contrats"  
  
FROM contrat ct  
  
LEFT JOIN region rg ON  
ct.Code_dep_code_commune =  
rg.Code_dep_code_commune  
  
GROUP BY reg_nom  
  
ORDER BY "Nb_Contrats" DESC
```

reg_nom character varying	Nb_Contrats bigint
Ile-de-France	13474
Provence-Alpes-Côte d'Azur	3287
Auvergne-Rhône-Alpes	2972
Nouvelle-Aquitaine	2097
Occitanie	1837
Hauts-de-France	1327
Pays de la Loire	1138
Bretagne	945
Normandie	898
Grand Est	806
Centre-Val de Loire	804
Bourgogne-Franche-Comté	402
Corse	247
Martinique	60
Guyane	37
La Réunion	4

Merci pour votre attention