

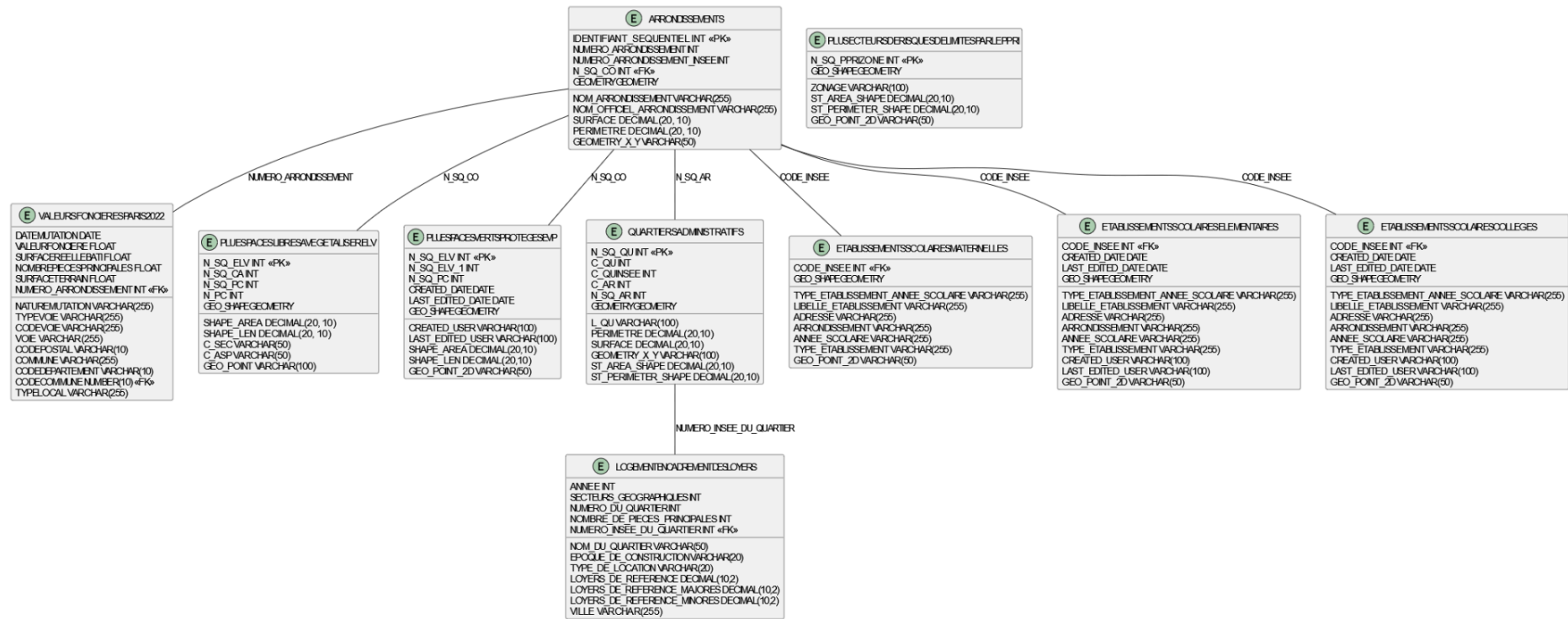
Rapport Etape 2

Analyse de l'immobilier à Paris

Introduction

Cette suite de projet explore la modélisation et la gestion de données à travers l'utilisation d'un diagramme entité-association (ER) et de scripts SQL. Le diagramme ER permet de visualiser les relations entre les entités, tandis que le premier script SQL définit la structure de la base de données. Par la suite, un script d'alimentation sera utilisé pour importer des données à partir de fichiers CSV dans la base de données. Cette approche garantit une gestion efficace des données et simplifie le développement de systèmes informatiques fiables.

1. Diagramme entité-association



Ce schéma représente un diagramme entité-association (ER) des différentes entités et de leurs relations dans une base de données pour la gestion des données immobilières et urbaines à Paris.

ARRONDISSEMENTS et quartiers_administratifs :

- Type de relation : "one-to-many"
- Clé étrangère : N_SQ_AR dans l'entité quartiers_administratifs, faisant référence à l'entité arrondissements.
- Signification : Chaque arrondissement peut avoir plusieurs quartiers administratifs associés.

ARRONDISSEMENTS et VALEURSFONCIERESPARIS2022 :

- Relation : One-To-Many (Un à Plusieurs)
- Signification : De Arrondissements vers ValeursFoncieresParis2022 (Un arrondissement peut avoir plusieurs valeurs foncières enregistrées pour différentes adresses, mais chaque valeur foncière est associée à un seul arrondissement).

QUARTIERSADMINISTRATIFS et LOGEMENTENCADREMENTDESLOYERS :

- Type de relation : "one-to-many"
- Clé étrangère : Numéro_INSEE_du_quartier dans l'entité logement_encadrement_des_loyers, faisant référence à l'entité quartiers_administratifs (N_SQ_QU).
- Signification : Chaque quartier administratif peut avoir plusieurs enregistrements de logements soumis à l'encadrement des loyers.

ARRONDISSEMENTS les deux entités PLUESPACESLIBRESAVEGETALISERELV et PLUESPACESVERTSPROTEGESEVP :

- Type de relation : "one-to-many"
- Clé étrangère : N_SQ_CA dans les entités correspondantes, faisant référence à l'entité arrondissements.
- Signification : Chaque arrondissement peut avoir plusieurs occurrences associées dans chacune des entités mentionnées.

ARRONDISSEMENTS et les entités

ETABLISSEMENTSSCOLAIRESMATERNELLES,
ETABLISSEMENTSSCOLAIRESELEMENTAIRES, et
ETABLISSEMENTSSCOLAIRESCOLLEGES:

- Type de relation : "one-to-many"
- Clé étrangère : Code_INSEE dans les entités correspondantes, faisant référence à l'entité arrondissements.

- Signification : Chaque arrondissement peut avoir plusieurs établissements scolaires maternelles, élémentaires et collèges associés mais un établissement n'a qu'un seul arrondissement.

L'indépendance de l'entité plu_secteurs_de_risques_delimites_par_le_ppri

Indépendante, elle offre une couche supplémentaire d'informations spécifiques aux risques d'inondation, permettant une évaluation plus complète des risques liés à l'emplacement des biens immobiliers, ce qui est crucial pour les investisseurs, les acheteurs, les planificateurs urbains et les compagnies d'assurance.

Dans l'ensemble, ces relations indiquent les liens entre les différentes entités de données, ce qui permet de modéliser efficacement les relations complexes entre les données dans le contexte de l'immobilier à Paris.

2. Création de la base de données

Ce script SQL `db_creation_script.sql` crée un schéma de base de données pour stocker des informations sur divers aspects de Paris. Il comporte plusieurs tables qui sont liées les unes aux autres de différentes manières.

Voici une explication optimisée du script avec une meilleure mise en évidence des relations entre les tables :

a. Table "VAEURSFONCIERESPARIS2022"

La table VAEURSFONCIERESPARIS2022 décrit les transactions foncières à Paris pour l'année 2022. Voici une description de ses colonnes :

- DATEMUTATION : La date de la mutation foncière.
- NATUREMUTATION : La nature de la mutation foncière.
- VAEURFONCIERE : La valeur foncière de la transaction.
- TYPEVOIE : Le type de voie où se trouve la propriété.
- CODEVOIE : Le code de la voie.
- VOIE : Le nom de la voie.
- CODEPOSTAL : Le code postal de la propriété.
- COMMUNE : La commune où se situe la propriété.
- CODEDEPARTEMENT : Le code du département où se situe la propriété.
- CODECOMMUNE : Le code de la commune où se situe la propriété.
- TYPELOCAL : Le type de local (appartement, maison, etc.).
- SURFACEREELLEBATI : La surface réelle du bâti.
- NOMBREPIECESPRINCIPALES : Le nombre de pièces principales.
- SURFACETERRAIN : La surface du terrain.
- NUMERO_ARRONDISSEMENT : Le numéro de l'arrondissement où se situe la propriété. Cette colonne est une clé étrangère faisant référence à la table ARRONDISSEMENTS, plus spécifiquement à la colonne NUMERO_ARRONDISSEMENT.

```
CREATE TABLE VAEURSFONCIERESPARIS2022 (  
  DATEMUTATION DATE,  
  NATUREMUTATION VARCHAR(255),  
  VAEURFONCIERE FLOAT,  
  TYPEVOIE VARCHAR(255),  
  CODEVOIE VARCHAR(255),  
  VOIE VARCHAR(255),  
  CODEPOSTAL VARCHAR(10),  
  COMMUNE VARCHAR(255),  
  CODEDEPARTEMENT VARCHAR(10),  
  CODECOMMUNE NUMBER(10),  
  TYPELOCAL VARCHAR(255),
```

```

SURFACEREELLEBATI FLOAT,
NOMBREPIECESPRINCIPALES FLOAT,
SURFACETERRAIN FLOAT,
NUMERO_ARRONDISSEMENT INT,
FOREIGN KEY (NUMERO_ARRONDISSEMENT) REFERENCES
ARRONDISSEMENTS(NUMERO_ARRONDISSEMENT)
);

```

b. Table "ARRONDISSEMENTS"

Cette requête SQL crée une table nommée ARRONDISSEMENTS avec les colonnes suivantes :

- IDENTIFIANT_SEQUENTIEL : Un identifiant séquentiel unique pour chaque enregistrement dans la table, défini comme une clé primaire (PRIMARY KEY).
- NUMERO_ARRONDISSEMENT : Le numéro de l'arrondissement.
- NUMERO_ARRONDISSEMENT_INSEE : Le numéro de l'arrondissement selon l'INSEE.
- NOM_ARRONDISSEMENT : Le nom de l'arrondissement.
- NOM_OFFICIEL_ARRONDISSEMENT : Le nom officiel de l'arrondissement.
- N_SQ_CO : Un numéro séquentiel de connexion.
- SURFACE : La surface de l'arrondissement, avec une précision de 20 chiffres au total et 10 chiffres après la virgule.
- PERIMETRE : Le périmètre de l'arrondissement, également avec une précision de 20 chiffres au total et 10 chiffres après la virgule.
- GEOMETRY_X_Y : Les coordonnées géographiques de l'arrondissement au format VARCHAR.
- GEOMETRY : Une colonne pour stocker la géométrie de l'arrondissement au format GEOMETRY.

```

CREATE TABLE ARRONDISSEMENTS (
    IDENTIFIANT_SEQUENTIEL INT PRIMARY KEY,
    NUMERO_ARRONDISSEMENT INT,
    NUMERO_ARRONDISSEMENT_INSEE INT,
    NOM_ARRONDISSEMENT VARCHAR(255),
    NOM_OFFICIEL_ARRONDISSEMENT VARCHAR(255),
    N_SQ_CO INT,
    SURFACE DECIMAL(20, 10),
    PERIMETRE DECIMAL(20, 10),
    GEOMETRY_X_Y VARCHAR(50),
    GEOMETRY GEOMETRY
);

```

c. Table LOGEMENTENCADREMENTDESLOYERS

Cette requête SQL crée une table nommée LOGEMENTENCADREMENTDESLOYERS avec les colonnes suivantes :

- ANNEE : Année associée aux données du logement.
- SECTEURS_GEOGRAPHIQUES : Numéro du secteur géographique.
- NUMERO_DU_QUARTIER : Numéro du quartier.
- NOM_DU_QUARTIER : Nom du quartier.
- NOMBRE_DE_PIECES_PRINCIPALES : Nombre de pièces principales dans le logement.
- EPOQUE_DE_CONSTRUCTION : Période de construction du logement.
- TYPE_DE_LOCATION : Type de location du logement.
- LOYERS_DE_REFERENCE : Loyers de référence pour le logement.
- LOYERS_DE_REFERENCE_MAJORES : Loyers de référence majorés.
- LOYERS_DE_REFERENCE_MINORES : Loyers de référence minorés.
- VILLE : Nom de la ville.
- NUMERO_INSEE_DU_QUARTIER : Numéro INSEE du quartier, avec une contrainte de clé étrangère qui référence la colonne N_SQ_QU de la table QUARTIERSADMINISTRATIFS.
- GEO_SHAPE : Géométrie associée au logement.
- GEO_POINT_2D : Coordonnées géographiques du logement au format VARCHAR.

```
CREATE TABLE LOGEMENTENCADREMENTDESLOYERS (  
  ANNEE INT,  
  SECTEURS_GEOGRAPHIQUES INT,  
  NUMERO_DU_QUARTIER INT,  
  NOM_DU_QUARTIER VARCHAR(50),  
  NOMBRE_DE_PIECES_PRINCIPALES INT,  
  EPOQUE_DE_CONSTRUCTION VARCHAR(20),  
  TYPE_DE_LOCATION VARCHAR(20),  
  LOYERS_DE_REFERENCE DECIMAL(10,2),  
  LOYERS_DE_REFERENCE_MAJORES DECIMAL(10,2),  
  LOYERS_DE_REFERENCE_MINORES DECIMAL(10,2),  
  VILLE VARCHAR(255),  
  NUMERO_INSEE_DU_QUARTIER INT,  
  GEO_SHAPE GEOMETRY,  
  GEO_POINT_2D VARCHAR(50),  
  FOREIGN KEY (NUMERO_INSEE_DU_QUARTIER) REFERENCES  
  QUARTIERSADMINISTRATIFS(N_SQ_QU)  
);
```


d. Table PLUESPACESLIBRESAVEGETALISERELV

Cette requête SQL crée une table nommée PLUESPACESLIBRESAVEGETALISERELV avec les colonnes suivantes :

- N_SQ_ELV : Numéro de séquence de l'espace libre à végétaliser.
- N_SQ_CA : Autre numéro de séquence associé à l'espace libre.
- N_SQ_PC : Numéro de séquence d'un code postal.
- SHAPE_AREA : Surface de l'espace libre à végétaliser.
- SHAPE_LEN : Longueur de l'espace libre à végétaliser.
- C_SEC : Catégorie de l'espace libre.
- C_ASP : Aspect de l'espace libre.
- N_PC : Numéro de code postal.
- GEO_SHAPE : Géométrie associée à l'espace libre à végétaliser.
- GEO_POINT : Point géographique de l'espace libre au format VARCHAR.

```
CREATE TABLE PLUESPACESLIBRESAVEGETALISERELV (  
  N_SQ_ELV INT,  
  N_SQ_CA INT,  
  N_SQ_PC INT,  
  SHAPE_AREA DECIMAL(20, 10),  
  SHAPE_LEN DECIMAL(20, 10),  
  C_SEC VARCHAR(50),  
  C_ASP VARCHAR(50),  
  N_PC INT,  
  GEO_SHAPE GEOMETRY,  
  GEO_POINT VARCHAR(100)  
);
```

e. Table PLUESPACESVERTSPROTEGESEVP

Cette requête SQL crée une table nommée PLUESPACESVERTSPROTEGESEVP avec les colonnes suivantes :

- N_SQ_ELV : Numéro de séquence du lieu vert protégé, défini comme clé primaire.
- N_SQ_ELV_1 : Autre numéro de séquence du lieu vert protégé.
- N_SQ_PC : Numéro de séquence d'un code postal.
- CREATED_USER : Utilisateur qui a créé l'enregistrement.
- CREATED_DATE : Date de création de l'enregistrement.
- LAST_EDITED_USER : Utilisateur qui a modifié pour la dernière fois l'enregistrement.
- LAST_EDITED_DATE : Date de la dernière modification de l'enregistrement.
- SHAPE_AREA : Surface du lieu vert protégé.
- SHAPE_LEN : Longueur du lieu vert protégé.
- GEO_SHAPE : Géométrie associée au lieu vert protégé.
- GEO_POINT_2D : Coordonnées géographiques du lieu vert protégé au format VARCHAR.

```
CREATE TABLE PLUESPACESVERTSPROTEGESEVP (
  N_SQ_ELV INT PRIMARY KEY,
  N_SQ_ELV_1 INT,
  N_SQ_PC INT,
  CREATED_USER VARCHAR(100),
  CREATED_DATE DATE,
  LAST_EDITED_USER VARCHAR(100),
  LAST_EDITED_DATE DATE,
  SHAPE_AREA DECIMAL(20,10),
  SHAPE_LEN DECIMAL(20,10),
  GEO_SHAPE GEOMETRY,
  GEO_POINT_2D VARCHAR(50)
);
```

f. Table PLUSECTEURSDERISQUESDELIMITESPARLEPPRI

Cette requête SQL crée une table nommée

PLUSECTEURSDERISQUESDELIMITESPARLEPPRI avec les colonnes suivantes :

- ZONAGE : Chaîne de caractères représentant le zonage.
- N_SQ_PPRIZONE : Numéro de séquence du secteur délimité par le PPRI, utilisé comme clé primaire.
- ST_AREA_SHAPE : Surface du secteur délimité par le PPRI.
- ST_PERIMETER_SHAPE : Périmètre du secteur délimité par le PPRI.
- GEO_SHAPE : Géométrie associée au secteur délimité par le PPRI.
- GEO_POINT_2D : Coordonnées géographiques du secteur délimité par le PPRI au format VARCHAR.

```
CREATE TABLE PLUSECTEURSDERISQUESDELIMITESPARLEPPRI (
  ZONAGE VARCHAR(100),
  N_SQ_PPRIZONE INT PRIMARY KEY,
  ST_AREA_SHAPE DECIMAL(20,10),
  ST_PERIMETER_SHAPE DECIMAL(20,10),
  GEO_SHAPE GEOMETRY,
  GEO_POINT_2D VARCHAR(50)
);
```

g. Table QUARTIERSADMINISTRATIFS

Cette requête SQL crée une table nommée QUARTIERSADMINISTRATIFS avec les colonnes suivantes :

- N_SQ_QU : Numéro de séquence du quartier, utilisé comme clé primaire.
- C_QU : Code du quartier.
- C_QUINSEE : Code INSEE du quartier.
- L_QU : Libellé du quartier.
- C_AR : Code de l'arrondissement auquel appartient le quartier.
- N_SQ_AR : Numéro de séquence de l'arrondissement.
- PERIMETRE : Périmètre du quartier.
- SURFACE : Surface du quartier.
- GEOMETRY_X_Y : Coordonnées géographiques du quartier au format VARCHAR.
- GEOMETRY : Géométrie associée au quartier.
- ST_AREA_SHAPE : Surface de la forme géométrique du quartier.
- ST_PERIMETER_SHAPE : Périmètre de la forme géométrique du quartier.

```
CREATE TABLE QUARTIERSADMINISTRATIFS (  
  N_SQ_QU INT PRIMARY KEY,  
  C_QU INT,  
  C_QUINSEE INT,  
  L_QU VARCHAR(100),  
  C_AR INT,  
  N_SQ_AR INT,  
  PERIMETRE DECIMAL(20,10)  
  SURFACE DECIMAL(20,10),  
  GEOMETRY_X_Y VARCHAR(100),  
  GEOMETRY GEOMETRY,  
  ST_AREA_SHAPE DECIMAL(20,10),  
  ST_PERIMETER_SHAPE DECIMAL(20,10)  
);
```

h. Table ETABLISSEMENTSSCOLAIRESMATERNELLES

Cette requête SQL crée une table nommée

ETABLISSEMENTSSCOLAIRESMATERNELLES avec les colonnes suivantes :

- TYPE_ETABLISSEMENT_ANNEE_SCOLAIRE : Type d'établissement scolaire pour l'année scolaire spécifiée.
- LIBELLE_ETABLISSEMENT : Libellé de l'établissement scolaire.
- ADRESSE : Adresse de l'établissement scolaire.
- ARRONDISSEMENT : Arrondissement où se situe l'établissement scolaire.
- CODE_INSEE : Code INSEE de l'arrondissement, avec une contrainte de clé étrangère qui référence la colonne N_SQ_CO de la table ARRONDISSEMENTS.
- ANNEE_SCOLAIRE : Année scolaire concernée par les données.
- TYPE_ETABLISSEMENT : Type d'établissement scolaire.

- GEO_SHAPE : Géométrie associée à l'établissement scolaire.
- GEO_POINT_2D : Coordonnées géographiques de l'établissement scolaire au format VARCHAR.

```
CREATE TABLE ETABLISSEMENTSSCOLAIRESMATERNELLES (
  TYPE_ETABLISSEMENT_ANNEE_SCOLAIRE VARCHAR(255),
  LIBELLE_ETABLISSEMENT VARCHAR(255),
  ADRESSE VARCHAR(255),
  ARRONDISSEMENT VARCHAR(255),
  CODE_INSEE INT,
  ANNEE_SCOLAIRE VARCHAR(255),
  TYPE_ETABLISSEMENT VARCHAR(255),
  GEO_SHAPE GEOMETRY,
  GEO_POINT_2D VARCHAR(50),
  FOREIGN KEY (CODE_INSEE) REFERENCES ARRONDISSEMENTS(N_SQ_CO)
);
```

i. Table ETABLISSEMENTSSCOLAIRESELEMENTAIRES

Cette requête SQL crée une table nommée ETABLISSEMENTSSCOLAIRESELEMENTAIRES avec les colonnes suivantes :

- TYPE_ETABLISSEMENT_ANNEE_SCOLAIRE : Type d'établissement scolaire pour l'année scolaire spécifiée.
- LIBELLE_ETABLISSEMENT : Libellé de l'établissement scolaire.
- ADRESSE : Adresse de l'établissement scolaire.
- ARRONDISSEMENT : Arrondissement où se situe l'établissement scolaire.
- CODE_INSEE : Code INSEE de l'arrondissement, avec une contrainte de clé étrangère qui référence la colonne N_SQ_CO de la table ARRONDISSEMENTS.
- ANNEE_SCOLAIRE : Année scolaire concernée par les données.
- TYPE_ETABLISSEMENT : Type d'établissement scolaire.
- CREATED_USER : Utilisateur ayant créé l'entrée dans la base de données.
- CREATED_DATE : Date de création de l'entrée dans la base de données.
- LAST_EDITED_USER : Dernier utilisateur ayant modifié l'entrée dans la base de données.
- LAST_EDITED_DATE : Date de la dernière modification de l'entrée dans la base de données.
- GEO_SHAPE : Géométrie associée à l'établissement scolaire.
- GEO_POINT_2D : Coordonnées géographiques de l'établissement scolaire au format VARCHAR.

```
CREATE TABLE ETABLISSEMENTSSCOLAIRESELEMENTAIRES (
  TYPE_ETABLISSEMENT_ANNEE_SCOLAIRE VARCHAR(255),
  LIBELLE_ETABLISSEMENT VARCHAR(255),
  ADRESSE VARCHAR(255),
  ARRONDISSEMENT VARCHAR(255),
```

```

CODE_INSEE INT,
ANNEE_SCOLAIRE VARCHAR(255),
TYPE_ETABLISSEMENT VARCHAR(255),
CREATED_USER VARCHAR(100),
CREATED_DATE DATE,
LAST_EDITED_USER VARCHAR(100),
LAST_EDITED_DATE DATE,
GEO_SHAPE GEOMETRY,
GEO_POINT_2D VARCHAR(50),
FOREIGN KEY (CODE_INSEE) REFERENCES ARRONDISSEMENTS(N_SQ_CO)
);

```

j. Table ETABLISSEMENTSSCOLAIRESCOLLEGES

Cette requête SQL crée une table nommée ETABLISSEMENTSSCOLAIRESCOLLEGES avec les colonnes suivantes :

- TYPE_ETABLISSEMENT_ANNEE_SCOLAIRE : Type d'établissement scolaire pour l'année scolaire spécifiée.
- LIBELLE_ETABLISSEMENT : Libellé de l'établissement scolaire.
- ADRESSE : Adresse de l'établissement scolaire.
- ARRONDISSEMENT : Arrondissement où se situe l'établissement scolaire.
- CODE_INSEE : Code INSEE de l'arrondissement, avec une contrainte de clé étrangère qui référence la colonne N_SQ_CO de la table ARRONDISSEMENTS.
- ANNEE_SCOLAIRE : Année scolaire concernée par les données.
- TYPE_ETABLISSEMENT : Type d'établissement scolaire.
- CREATED_USER : Utilisateur ayant créé l'entrée dans la base de données.
- CREATED_DATE : Date de création de l'entrée dans la base de données.
- LAST_EDITED_USER : Dernier utilisateur ayant modifié l'entrée dans la base de données.
- LAST_EDITED_DATE : Date de la dernière modification de l'entrée dans la base de données.
- GEO_SHAPE : Géométrie associée à l'établissement scolaire.
- GEO_POINT_2D : Coordonnées géographiques de l'établissement scolaire au format VARCHAR.

```

CREATE TABLE ETABLISSEMENTSSCOLAIRESCOLLEGES (
    TYPE_ETABLISSEMENT_ANNEE_SCOLAIRE VARCHAR(255),
    LIBELLE_ETABLISSEMENT VARCHAR(255),
    ADRESSE VARCHAR(255),
    ARRONDISSEMENT VARCHAR(255),
    CODE_INSEE INT,
    ANNEE_SCOLAIRE VARCHAR(255),
    TYPE_ETABLISSEMENT VARCHAR(255),
    CREATED_USER VARCHAR(100),

```

```
    CREATED_DATE DATE,  
    LAST_EDITED_USER VARCHAR(100),  
    LAST_EDITED_DATE DATE,  
    GEO_SHAPE GEOMETRY,  
    GEO_POINT_2D VARCHAR(50),  
    FOREIGN KEY (CODE_INSEE) REFERENCES ARRONDISSEMENTS(N_SQ_CO)  
);
```

