

RAPPORT

Projet de Certification Développeur Data

Maëlle Coriou

Promotion Dev Data : 2020/2021

Ecole de formation : Simplon Nantes

I - INTRODUCTION	2
II - COMPRÉHENSION DE LA DEMANDE CLIENT	3
1 - Objectifs et enjeux	3
2 - Les besoins	4
3 - L'état de l'art	5
4 - Traduction métier du besoin client	6
III - MISE EN OEUVRE DU PROJET	7
1 - Gestion du projet	7
2 - Organisation du projet	8
3 - Langages et outils	8
4 - Collecte des données	9
5 - Description des données sélectionnées	12
6 - Modélisation de la base de données	14
7 - Préparation des données	20
8 - Sauvegarde des fichiers	25
IV - INTERROGATION DE LA BASE DE DONNEES	26
1 - Requêtes Sql	26
2 - Analyse des données sous représentations graphiques	28
V - CONCLUSION	30

I - INTRODUCTION

En raison de la pandémie COVID-19, les musées et le secteur culturel sont parmi les plus touchés.

Aujourd'hui, plus que jamais, les musées sont confrontés à des défis uniques liés aux questions sociales, économiques et écologiques. Tout en étant les témoins du passé et les gardiens des trésors de l'humanité pour les générations futures, les musées jouent un rôle clé dans le développement par l'éducation et la démocratisation.

En attendant leur réouverture prochaine, j'ai eu l'idée de développer mon projet de certification sur le thème de l'art et de notre patrimoine.

Le gouvernement nous donne accès libre à l'inventaire des œuvres exposées dans nos musées ayant reçu l'appellation « Musée de France ».

L'Appellation « Musée de France » a été créée par la loi du 4 janvier 2002.

Ainsi est considéré comme « Musée de France », au sens de cette loi, « toute collection permanente composée de biens dont la conservation et la présentation revêtent un **intérêt** public et organisée en vue de la connaissance, de l'éducation et du plaisir du public » (Art. L. 410-1.).

Un « Musée de France » labellisé, c'est un bâtiment, des collections accessibles au public, une équipe de professionnels, une garantie de qualité.

La France compte un peu plus de 1.200 institutions ayant reçu cette appellation. Répartis sur l'ensemble du territoire national, les musées de France sont d'une grande variété.

Afin de valoriser cette mine d'informations mises à notre disposition, je me suis projetée dans un contexte client : quels seraient les besoins auxquels cette source pourrait répondre ?

Face au contexte actuel, le gouvernement allège progressivement les restrictions, nous pouvons désormais nous déplacer, c'est annoncé, les musées vont rouvrir !
Il nous tarde de faire des sorties culturelles et de sillonner les routes.

Alors, j'ai imaginé une entreprise dont l'objectif commercial pourrait répondre à ce besoin.

Je l'ai nommé : **Go Explore**.

II - COMPRÉHENSION DE LA DEMANDE CLIENT

1 - Objectifs et enjeux

Go Explore est un tour opérateur.

Il organise des voyages clé en main sur différents thèmes :

Séjours d'aventures en Afrique, ascensions des plus beaux sommets du monde, traversée de l'Europe à vélo...

Également touché par la crise Covid, conscient que cette crise a fortement ébranlé le domaine du tourisme Français, son objectif est de participer à redonner une dynamique au secteur.

Il souhaite donc renouveler son offre en proposant des circuits sur le thème de l'Art.

L'objectif est de permettre au voyageur de sillonner la France, de musées en musées sur le thème de son choix.

Voyageur passionné ou souhaitant développer ses connaissances sur un artiste, un mouvement artistique, une époque; le séjour pourra être organisé en fonction des musées exposant ces œuvres.

Son offre propose également des nuitées dans des hôtels indépendants labellisés hôteliers de France, une découverte de la gastronomie régionale avec les acteurs locaux auprès desquels il aura développé un partenariat.

Pour se faire, le directeur de **Go Explore** souhaite avoir accès aux données des musées de France afin de développer cette nouvelle offre.

L'objectif est donc de développer un outil lui permettant l'accès et l'analyse des données des musées de France sur l'ensemble des œuvres exposées afin de réaliser des séjours clé en main.

Cahier des charges validé pour un livrable au 19/05/2021.

2 - Les besoins

Données principales nécessaires :

- Informations musées :
adresse, numéro de téléphone, géolocalisation, Url site, description
- Informations inventaires :
Artistes, titres, sujets, thématiques, époques, matériaux, techniques

Informations et analyses attendues :

Cibler les lieux d'expositions :
recherches par Artistes, titres d'œuvres, techniques, périodes

Analyse par artistes :
nombre d'œuvres
nombre de musées exposants
nombre d'œuvres exposées par musées
localisation des musées exposants

Analyse par musées :
nombre d'artistes
nombre d'œuvres par époque
nombre d'œuvres par techniques et matériaux
nombre d'œuvres par domaine

Analyse par région :
domaines principaux, artistes phares, époques, nombre de musées, distance

3 - L'état de l'art

Des étudiants de l'IUT de Tours ont développé des cartographies dynamiques basées sur les données des inventaires de collections des musées de France, extrait de la base Joconde.

Datavisualisation permettant aux étudiants de l'histoire et de l'art de mieux connaître le patrimoine de leur région selon les périodes et techniques.

La réalisation permet entre autre de répondre aux problématiques suivantes :

- ❖ [Les sujets des oeuvres](#)
- ❖ [Quels sont les principaux matériaux ?](#)
- ❖ [Combien d'oeuvres pour chaque époque ?](#)
- ❖ [Carte par matériaux](#)
- ❖ [Carte par sujets](#)
- ❖ [Carte par époque](#)
- ❖ [Cartographie des musées de France](#)

Travaux publiés en mars 2019.

POP : la plateforme ouverte du patrimoine

La plateforme POP regroupe les contenus numériques du patrimoine français afin de les rendre accessibles et consultables au plus grand nombre

Les musées de France publient les données (notices illustrées descriptives d'objets) sur Joconde, catalogue collectif des collections des musées de France.

Depuis 2019, les musées sont autonomes pour créer et modifier les notices grâce à POP, plateforme ouverte du patrimoine.

Le versement sur la plateforme de production de POP a été délibérément simplifié afin d'encourager les musées de France à intégrer la publication sur Joconde comme une étape naturelle de la gestion de leurs collections.

Recherches possibles par domaine, artiste, lieu, période, technique.

Datavisualisation sous liste, carte ou mosaïque.

La plateforme POP diffuse des images et textes soumis aux droits d'auteur.

[Lien vers le site pop.culture.gouv.fr](https://pop.culture.gouv.fr)

[Lien vers droits d'utilisation de la plateforme POP](#)

4 - Traduction métier du besoin client

Périmètre du projet :

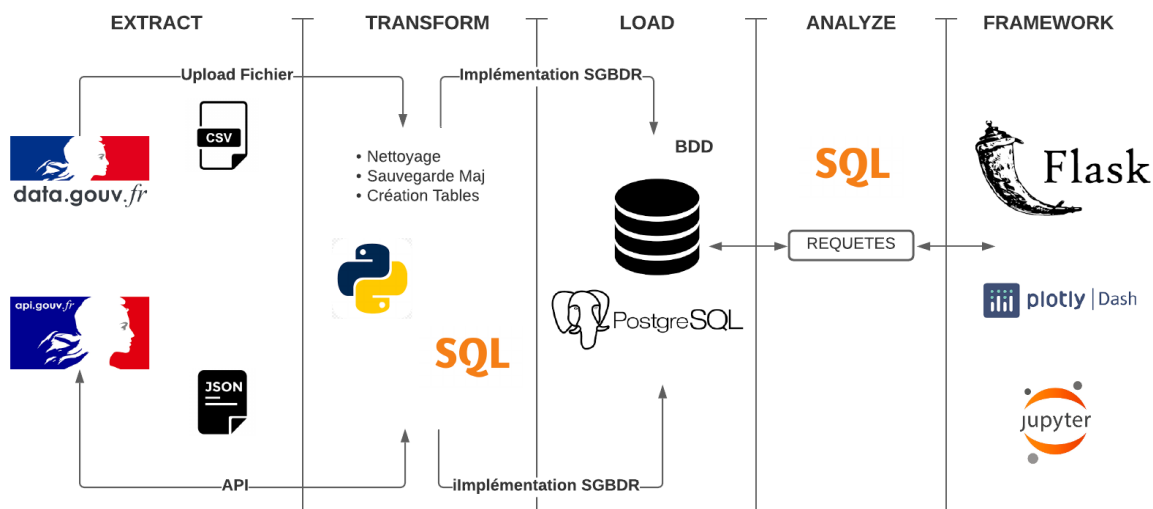
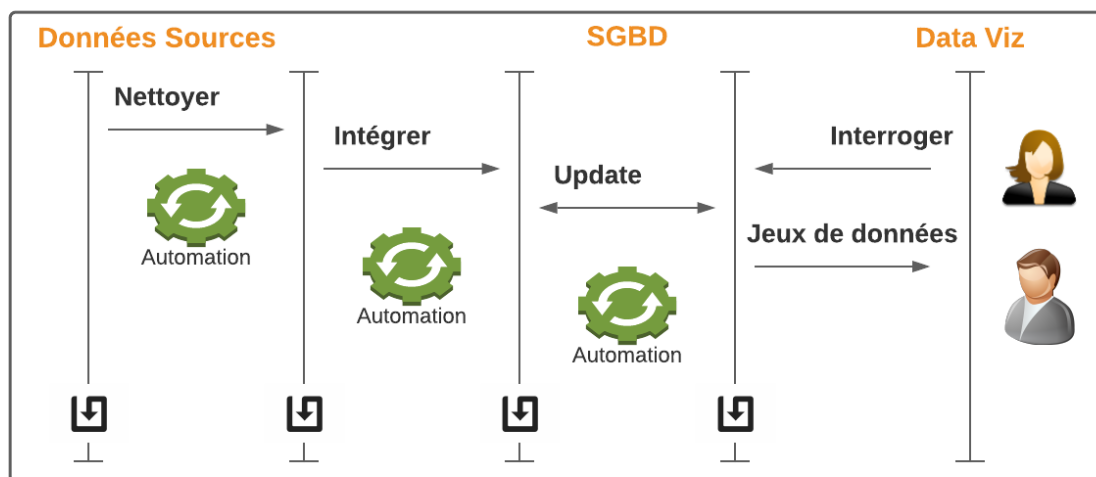
Développement d'une base de données relationnelle sous Postgresql :

- Import fichiers CSV extraits de la base Joconde
- Préparation des données en vue de leur intégration
- Création d'une base de données socle
- Update BDD automatisée via requêtes API de la base Joconde

Analyse BDD sous requêtes SQL et valorisation du contenu selon les besoins :

- Datavisualisation : Bibliothèques Python : Plotly et Folium, Frameworks Dash et Flask

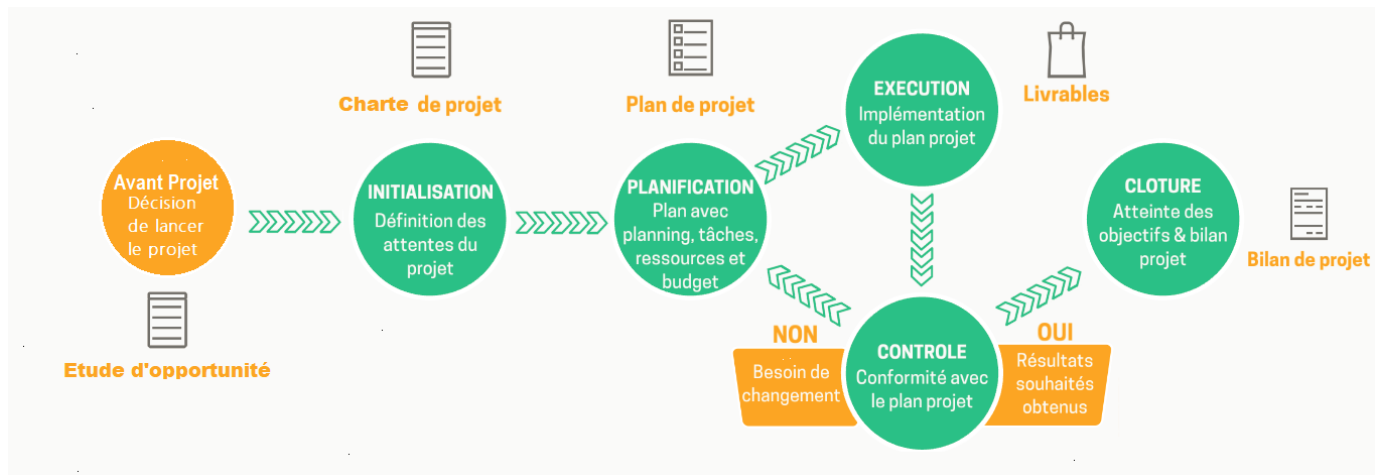
Schémas fonctionnel et logiciel du projet :



III - MISE EN OEUVRE DU PROJET

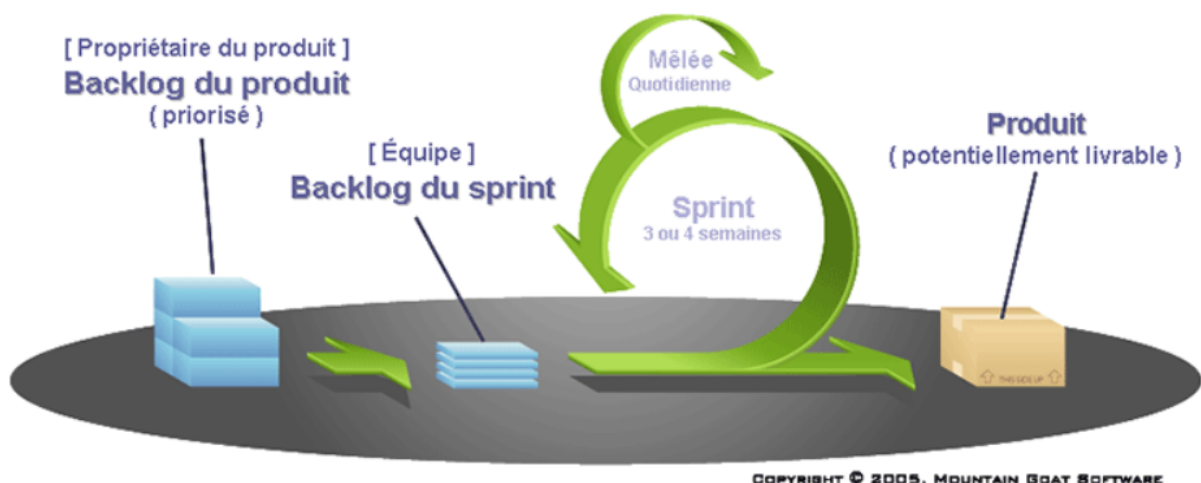
1 - Gestion du projet

Les étapes que j'ai suivi pour la réalisation de ce projet :



J'ai choisi ce schéma pour mettre en avant la manière dont j'ai avancé sur ce projet. Inspirée de la méthode agile, j'ai défini mes objectifs sur plusieurs phases, afin d'avoir une fonctionnalité développée à chaque phase. En revanche, je n'ai pas eu les phases de daily meeting ni de rétro en fin de sprint sur mon projet faute d'équipe et de client. Cependant, j'avais des points 2 à 3 fois par semaine avec mon formateur.

Je devrais avoir une 3ème version semaine du 19 mai si tout se passe bien.

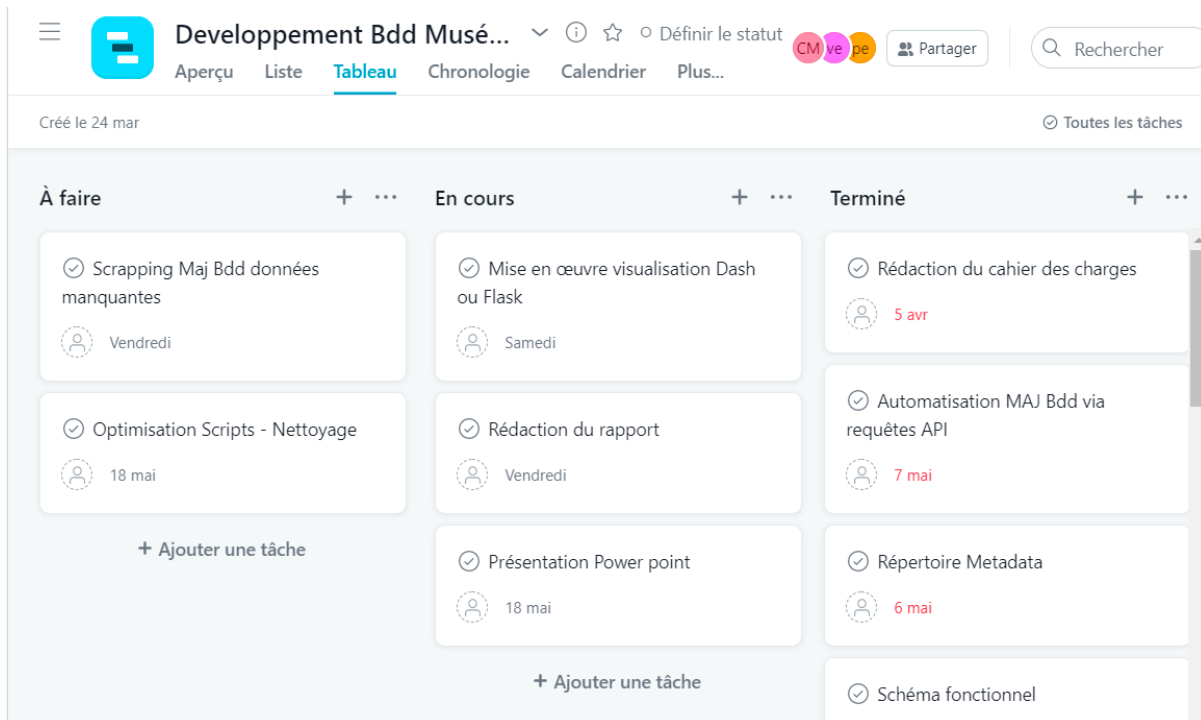


[Le cycle de vie d'un projet de construction](#)

2 - Organisation du projet

Pour le planning et l'organisation du projet, j'ai utilisé la méthode Kanban en mettant en place un workflow des tâches à réaliser avec leurs priorités et deadlines.

J'ai utilisé l'application Asana, voici un aperçu de mon tableau de bord :



3 - Langages et outils

Voici le récapitulatif des principaux langages, librairies et outils utilisés pour la mise en oeuvre du projet :

Langages :

python==3.8.6
PostgreSql==4.28

Librairies :

dash==1.20.0
Flask==1.1.2
folium==0.12.1
jupyterlab==2.2.9
matplotlib==3.3.2
numpy==1.19.3
pandas==1.1.3
plotly==4.14.3
psycopg2==2.8.6
pymongo==3.11.2
requests==2.24.0
schedule==1.0.0
SQLAlchemy==1.3.20

Outils :

Dbeaver 21.0.0
Git Bash
Jupyter NoteBook
MongoDb Compass
Pg Admin 4.28 pour PostgreSQL
Visual Studio Code

4 - Collecte des données

Afin de répondre à la problématique client, 2 jeux de données sont nécessaires à la réalisation du projet :

1- Collections des musées de France : extrait de la base Joconde

La base Joconde décrit les œuvres des musées de France.
650k Enregistrements.

Upload fichiers formats Csv, Json ou via Api format json.

[Collections des musées de France : extrait de la base Joconde](#)

Jeu de données sous licence ouverte : [Licence Ouverte Version 2.0 \(etalab.gouv.fr\)](#)

Il est possible d'importer la totalité du jeu de données ou de faire une sélection personnalisée par mot clé, domaine ou ville.

Voici un aperçu des champs et données du dataset :

Nom Champ	Type	Exemple information
Référence	texte	M5037010624
Ancien dépôt	texte	
Appellation	texte	
Auteur	texte	DANTAN Jean Pierre;DANTAN le Jeune (dit)
Date acquisition	texte	1888 vers
Date de dépôt	texte	1928 déposé
Découverte / collecte	texte	
Dénomination	texte	buste
Lieu de dépôt	texte	en dépôt;Paris;musée du Louvre département des Sculptures
Mesures	texte	H. 23.5 (dont piédouche : 5.5) ; L. 13 ; 10 Pr
Date de mise à jour	date	
Date de création	date	2001-02-22
Domaine	texte	sculpture
Date sujet représenté	texte	1758 né ; 1836 mort
Ecole-pays	texte	France
Epoque	texte	
Lieu historique	texte	
N°Inventaire	texte	RF 1965
Appellation « musée de France »	texte	Musée de France#au sens de la loi n°2002-5 du 4 janvier 2002
Lieu de création/utilisation	texte	Italie;Rome (lieu d'exécution)
Localisation	texte	Paris ; musée du Louvre département des Sculptures
Millésime de création	texte	1829

Nom Champ	Type	Exemple information
Millésime d'utilisation	texte	
Identifiant Museofile	texte	M5031
Nom officiel du musée	texte	musée du Louvre
Onomastique	texte	
Précision auteur	texte	Paris, 1800 ; Bade, 1869
Période de l'original copié	texte	
Période de création	texte	2e quart 19e siècle
Période d'utilisation	texte	
Région	texte	Ile-de-France
Sujet représenté	texte	portrait charge (Vernet Horace, homme, en buste, satirique, peintre, dessinateur)
Source représentation	texte	
Statut juridique	texte	propriété de la commune;legs;Paris, musée Carnavalet
Matériaux – techniques	texte	bronze
Titre	texte	Carle Vernet, Charles-Horace, dit (Bordeaux, 1758-Paris, 1836)
Utilisation / Destination	texte	
Ville	texte	Paris
Lien site associé	texte	
geolocalisation_ville	geo_point_2d	[48.8589,2.3469]

2- Musées de France : base Muséofile

La base Muséofile répertorie les musées bénéficiant de l'appellation « musée de France »

1.220 Enregistrements,

Upload fichiers formats Csv, Json ou via Api format json.

[Musées de France : base Muséofile — Ministère de la Culture](#)

Jeu de données sous licence ouverte : [Licence Ouverte Version 2.0 \(etalab.gouv.fr\)](#)

Voici un aperçu des champs et données du dataset :

Nom Champ	Type	Exemple information
Identifiant	texte	M0144
Adresse	texte	12, rue Camille Rodier
Artiste	texte	Théodore Lévine (1848 - 1912) ; Paul Jean Baptiste Gasq (1860 - 1944) ; Jean François (1906 - 1980).
Atout	texte	Collections archéologiques gallo-romaines (site des Bolards). Collections mérovingiennes. Collections militaires : les chasseurs à pied, armes, costumes et souvenirs évoquant la bataille de Nuits (18/12/1870) . Peintures de Jean François sur la vigne et le vin.
Catégorie	texte	Musée de site
Code Postal	texte	21700
Domaine thématique	texte	Archéologie;Beaux-arts;Ethnologie;Histoire
Département	texte	Côte-d'Or
Date de saisie	texte	2019-01-21
Histoire	texte	Le docteur Duret (1794-1874), maire de la ville, jette les bases du premier musée en créant en 1846 un cabinet d'archéologie et d'histoire naturelle. Le capitaine J. Derosne fonde le musée des Chasseurs. Ce musée (rebaptisé musée Driant) rassemble armes et uniformes ainsi que des pièces en rapport avec la guerre de 1870. Cette collection s'est enrichie de tenues militaires de célèbres Nuits. En 1971, dons de médailles et de monnaies. En 1975, le docteur Ernest Planson fonde le musée actuel en installant dans les caves de la Maison Rodier les collections gallo-romaines issues du site des Bolards.
Intérêt	texte	Ancienne maison de vins appartenant à la famille de Camille Rodier, co-fondateur de la Confrérie des Chevaliers du Tastevin.
Lieu	texte	Ancienne Maison de Vins
Nom officiel	texte	musée municipal
Nom usage	texte	Musée de Nuits-Saint-Georges
Personnage phare	texte	
Protection bâtiment	texte	
Protection espace	texte	Aux abords d'un monument historique
Région	texte	Bourgogne-Franche-Comté
Téléphone	texte	03 80 62 01 37
Thèmes	texte	Archéologie nationale : Gallo-romain, Paléo-chrétien (Mérovingien);Beaux-Arts : Estampe et Affiche;Collections militaires : Armes, Uniformes, Armures;Numismatique;Histoire (Cîteaux), Histoire militaire;Musique : Instruments;Sciences de la nature : Botanique, Géologie
URL	texte	www.ot-nuits-st-georges.fr/fr/fiche.html?id_fiche=2437
Ville	texte	Nuits-Saint-Georges
geolocalisation	geo_point_2d	[47.138661,4.950405]

5 - Description des données sélectionnées

Les jeux de données ont un large éventail d'informations. Cependant, pour le projet, toutes ne seront pas nécessaires pour répondre aux attentes client.

Le jeu d'inventaire permettra de collecter les informations sur les œuvres, le jeu Muséofile apportera la localisation des musées ainsi que des informations sur leurs atouts, caractéristiques, Url site web...

Pour la première version de la base de données, j'ai sélectionné l'inventaire des œuvres du domaine Peintures.

Cela me permet dans un premier temps de travailler sur un dataset de 75k enregistrements et de pouvoir faire valider une première version fonctionnelle au client avant d'implanter le reste des données nécessitant une préparation et un nettoyage important avant leur intégration dans la base de données.

Cette première version pourra alors être testée et évaluée par le client. Des modifications et optimisations pourront alors être mises en place pour la seconde version.

Après extraction du csv, j'ai procédé à une première évaluation des données en utilisant la bibliothèque Pandas afin de lire le fichier Csv sous dataframe et pouvoir procéder à quelques analyses.

Inventaire Oeuvres Domaine Peintures :

df.shape : (75403 lignes, 40 colonnes)

df.isnull().sum() : somme des valeurs nulles

Référence	35100	Localisation	6
Ancien dépôt	74941	Millésime de création	51882
Appellation	75077	Millésime d'utilisation	75297
Auteur	303	Identifiant Museofile	8
Date acquisition	10896	Nom officiel du musée	35193
Date de dépôt	71832	Onomastique	75364
Découverte / collecte	75171	Précision auteur	39325
Dénomination	42591	Période de l'original copié	74455
Lieu de dépôt	71190	Période de création	32731
Mesures	5612	Période d'utilisation	75231
Date de mise à jour	34667	Région	31509
Date de création	35292	Sujet représenté	41533
Domaine	0	Source représentation	73480
Date sujet représenté	69445	Statut juridique	232
Ecole-pays	10760	Matériaux – techniques	35392
Epoque	73544	Titre	118
Lieu historique	75352	Utilisation / Destination	71863
N°Inventaire	31414	Ville	104
Appellation « musée de France »	36222	Lien site associé	57747
Lieu de création/utilisation	69265	geolocalisation_ville	279

Ce premier résultat me permet de constater que beaucoup de champs ne sont pas renseignés par les musées. Je vais pouvoir exclure un certain nombre de colonnes.

Cependant, il me semble judicieux de conserver certaines informations même si elles ne sont pas complètes. Celles-ci pourront cependant être complétées dans une phase de développement secondaire grâce à un scraping du site wikipédia par exemple. La phase de nettoyage et la suppression des doublons minimiseront également ces valeurs nulles.

Choix de sélection : 17 colonnes

Référence	35100	Lieu de création/utilisation	69265
Auteur	303	Millésime de création	51882
Date acquisition	10896	Identifiant Museofile	8
Dénomination	42591	Précision auteur	39325
Date de mise à jour	34667	Période de création	32731
Date de création	35292	Sujet représenté	41533
Domaine	0	Matériaux – techniques	35392
Ecole-pays	10760	Titre	118
Epoque	73544		

Les musées :

df.shape : (1218 lignes, 23 colonnes)

df.isnull().sum() : somme des valeurs nulles

ref	0	nomoff	2
adrl1_m	68	nomusage	487
artiste	776	phare	659
atout	91	prot_bat	779
categ	960	prot_esp	670
cp_m	1	region	1
dompal	38	tel_m	27
dpt	0	themes	133
dt_saisi	39	url_m	138
hist	105	ville_m	0
interet	353	geolocalisation	96
lieu_m	704		

Concernant ces données, j'ai pris la décision de conserver la totalité. Selon les requêtes et résultats, je pourrai prendre la décision de supprimer certaines colonnes.

Les informations manquantes pourront également être complétées ultérieurement.

6 - Modélisation de la base de données

J'ai fait le choix de développer une base de données type relationnelle pour sa scalabilité. L'augmentation du volume de données stocké n'a pas d'incidence sur les données existantes et l'organisation de la base.

Les SGBDR facilitent également l'automatisation des process.

Les données sont stockées et extraites aisément grâce aux requêtes SQL.

[12 règles de Codd — Wikipédia](#)

[Propriétés ACID — Wikipédia](#)

[CRUD — Wikipédia](#)

Pour la réalisation de la modélisation de la base de données, j'ai utilisé le logiciel Looping. Logiciel gratuit et libre d'utilisation développé par l'Université Toulouse III.

Cet outil permet de créer un premier schéma des entités et de leurs associations.

Les schémas UML et MLD sont produits automatiquement sur la base de la première modélisation entités / associations.

Lien de téléchargement du logiciel : [Looping - Modélisation Conceptuelle de Données](#)

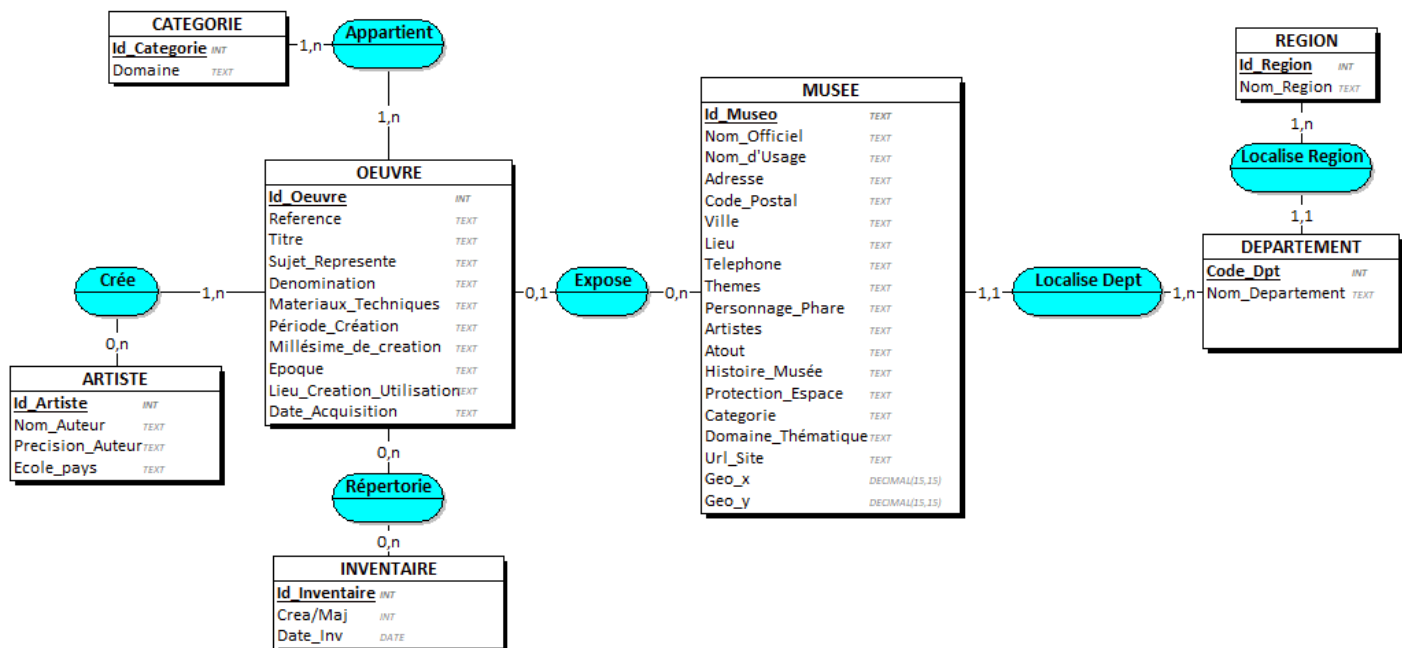
Quelques mots sur l'architecture d'une base de données :

Une base de données relationnelle repose sur ses tables, chaque table a ses attributs, pour chaque attribut il est nécessaire d'en déterminer son type. Enfin, on détermine le type de relation entre chaque table grâce à leurs identifiants.

L'étape de modélisation des entités et leurs relations est cruciale avant de se lancer dans la création des tables de la base de données ceci afin de s'assurer la cohérence entre chaque table et une insertion des données fluide.

Ressource : [Petit cours de modélisation - C.Darmangeat](#)

Phase 1 : schéma MCD Entités / Relations retenu :



Modélisation structurée sur 7 entités :

2 entités principales :

MUSEE : 17 attributs

OEUVRE : 10 attributs

5 entités secondaires :

REGION : 1 attribut

DEPARTEMENT : 1 attribut

CATEGORIE : 1 attribut

ARTISTE : 3 attributs

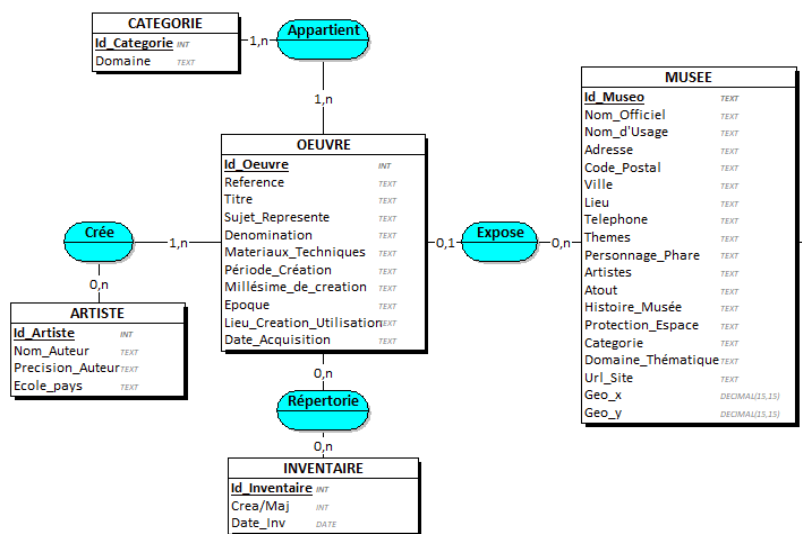
INVENTAIRE : 2 attributs

J'ai choisi de mettre en place ces 5 entités secondaires afin d'éviter la redondance des données.

Les tables centrales auront peu de valeurs répétées.

Description des relations et de leurs cardinalités :

Relations de l'entité Oeuvre :



Oeuvre - Catégorie : Many to Many

- 1 à plusieurs oeuvres appartiennent à 1 à plusieurs catégories
- 1 à plusieurs catégories appartiennent à 1 à plusieurs oeuvres

Oeuvre - Artiste : Many to Many

- 1 à plusieurs oeuvres ont été créés par 1 à plusieurs artistes
- 1 à plusieurs artistes ont créé 0 à plusieurs oeuvres

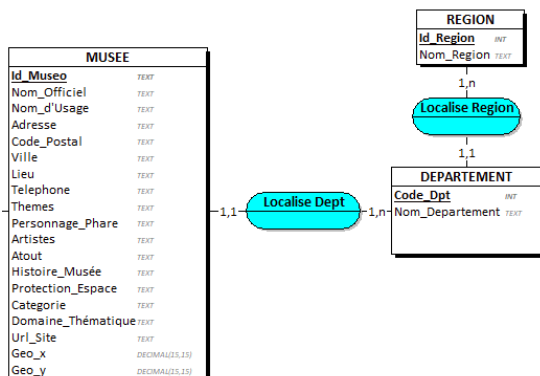
Oeuvre - Inventaire : Many to Many

- 0 à plusieurs oeuvres ont été répertoriées lors d'un inventaire à
- 0 ou plusieurs dates : Création de fiche ou mise à jour de fiche

Oeuvre - Musée : One to Many

- 0 à plusieurs oeuvres sont exposées dans un musée
- 0 à plusieurs musées exposent une à plusieurs oeuvres

Relations de l'entité Musée :



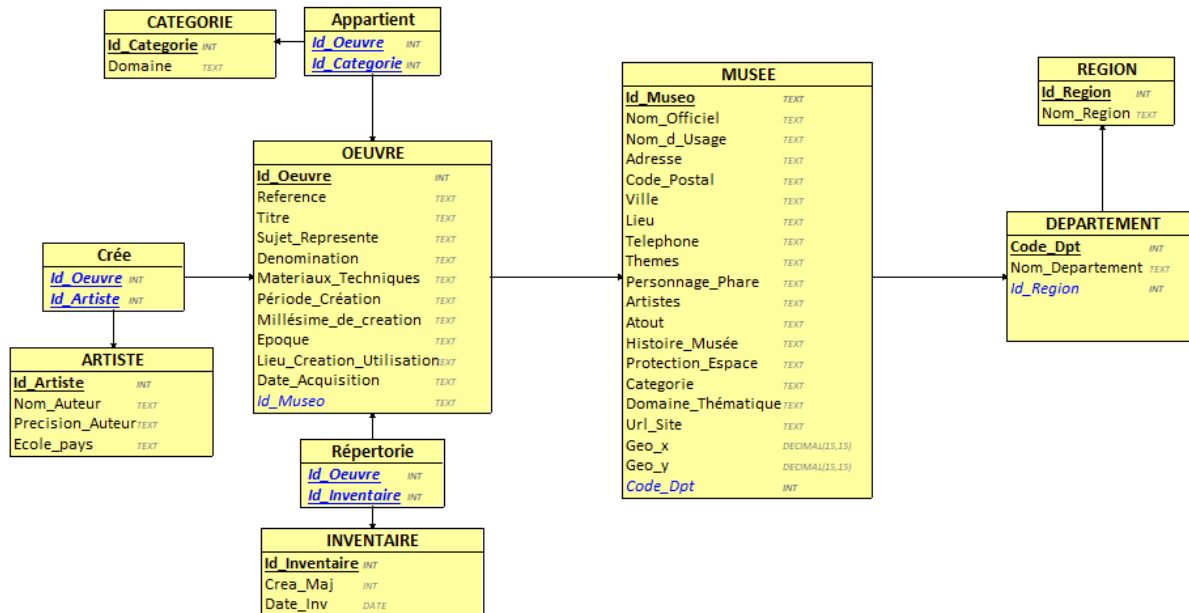
Musée - Département : One to Many

- 0 à plusieurs Musées sont localisés dans 1 Dpt
- 1 Dpt a 1 à plusieurs musées

Département - Région : One to Many

- 1 à plusieurs Dpt sont localisés dans 1 région
- 1 région a un à plusieurs Dpt

Phase 2 : schéma MLD Merise :



Ce schéma permet de définir les identifiants faisant le lien entre chaque table.

Id_Museo : One to Many - OEUVRE / MUSEE

Après analyse des données, je constate que je vais pouvoir me servir de l'Id Museo, identifiant unique d'un musée.

Ce code est renseigné dans le fichier des collections pour chaque œuvre, il est également présent dans le fichier Muséofile pour chaque Musée.

Cet Id sera donc la clé primaire de la table Musée et clé étrangère de la table œuvre.

Détail des tables :

Id_Dpt : One to Many - MUSEE / DEPARTEMENT

Pour cette relation, je vais devoir créer une nouvelle colonne code département dans le fichier source car cette information n'y figure pas. Extraction des 2 premiers chiffres du code postal présent dans le fichier source.

Cet Id sera la clé primaire de la table Département et clé étrangère de la table Musée.

Id_Region : One to Many - DEPARTEMENT / REGION

Même contrainte que l'Id_Dpt, nécessité d'insérer une nouvelle colonne avec le code région. Pour cette information, je vais récupérer ces codes sur le site [Codes géographiques de la France — Wikipédia](#)

j'ajouterai les codes manuellement au fichier qui me servira à l'insertion des données.

Cet Id sera la clé primaire de la table Région et clé étrangère de la table Département.

Id_Oeuvre / Id_Categorie : Many to Many - OEUVRE / CATEGORIE




La complexité que je rencontre dans le jeu de données est que l'information de la colonne Domaine est multiple et redondante à chaque enregistrement. D'où la nécessité de créer une table listant ces informations de manière unique, informations nécessaires aux besoins du projet.

Exemple :

peinture;histoire
peinture
peinture;marine

Après extraction de chaque valeur avec la méthode split, list et suppression des doublons, l'intégration de chaque domaine sera possible à la table Catégorie, un Id unique sera affecté avec une contrainte de valeur unique au champ nom_domaine.

Une table de jonction sera nécessaire pour lier la table œuvre à la table Catégorie. Une œuvre pourra appartenir à plusieurs catégories et inversement. La table Id_oeuvre / Id_categorie listera toutes relations existantes entre une Id oeuvre avec une Id catégorie.

Name	Type	Table	Object
 oeuvre_categorie_id_oeuvre_fkey	Foreign Key	oeuvre_categorie	oeuvre_categorie_id_oeuvre_fkey
 oeuvre_categorie_id_categorie_fkey	Foreign Key	oeuvre_categorie	oeuvre_categorie_id_categorie_fkey
 oeuvre_categorie_pkey	Primary Key	oeuvre_categorie	oeuvre_categorie_pkey

Id_Oeuvre / Id_Artiste : Many to Many - OEUVRE / ARTISTE

Je rencontre la même complexité que pour les domaines. Plusieurs artistes ont travaillé sur la même œuvre, un artiste a réalisé plusieurs œuvres mais a pu collaborer avec d'autres artistes pour d'autres œuvres. Un artiste peut avoir un nom d'utilisateur (dit).

Exemple :

LAFAYE Prosper;ALAUX Jean;LE ROMAIN (dit)
LECOMTE Hippolyte;ALAUX Jean;LE ROMAIN (dit)
LAFAYE Prosper;ALAUX Jean;LE ROMAIN (dit);TAUNAY Nicolas Antoine (d'après)

Je mettrai donc en place la même méthode que pour les domaines.




La table de jonction Id_oeuvre / Id_artiste listera toutes relations existantes entre une Id œuvre avec une Id artiste.

Name	Type	Table	Object
 oeuvre_artiste_id_oeuvre_fkey	Foreign Key	oeuvre_artiste	oeuvre_artiste_id_oeuvre_fkey
 oeuvre_artiste_id_artiste_fkey	Foreign Key	oeuvre_artiste	oeuvre_artiste_id_artiste_fkey
 oeuvre_artiste_pkey	Primary Key	oeuvre_artiste	oeuvre_artiste_pkey

Id_Oeuvre / Id_Inventaire : Many to Many - OEUVRE / INVENTAIRE

Cette relation Many-to-Many est nécessaire dans le sens où nous avons dans le jeu de données une date de création indiquant la date à laquelle l'inventaire a été fait (date de la création de la fiche de l'œuvre). Seconde indication: la date à laquelle la fiche a été mise à jour.

J'ai souhaité créer cette table afin d'avoir un indicateur de date d'enregistrement ou de mise à jour des œuvres afin d'interroger la base Joconde et extraire les éventuelles mises à jour vs la date enregistrée dans la Bdd.

Name	Type	Table	Object
 oeuvre_inventaire_id_oeuvre_fkey	Foreign Key	oeuvre_inventaire	oeuvre_inventaire_id_oeuvre_fkey
 oeuvre_inventaire_id_inventaire_fkey	Foreign Key	oeuvre_inventaire	oeuvre_inventaire_id_inventaire_fkey
 oeuvre_inventaire_pkey	Primary Key	oeuvre_inventaire	oeuvre_inventaire_pkey

Exemples des contraintes mises en place à la création des tables :

Table Oeuvre_Artiste, Many to Many :

```
CREATE TABLE public.oeuvre_artiste (  
  id_oeuvre int4 NOT NULL,  
  id_artiste int4 NOT NULL,  
  CONSTRAINT oeuvre_artiste_pkey PRIMARY KEY (id_oeuvre, id_artiste),  
  CONSTRAINT oeuvre_artiste_id_artiste_fkey FOREIGN KEY (id_artiste) REFERENCES artiste(id_artiste),  
  CONSTRAINT oeuvre_artiste_id_oeuvre_fkey FOREIGN KEY (id_oeuvre) REFERENCES oeuvre(id_oeuvre)  
);
```

Table Département, One to Many :

```
CREATE TABLE public.departement (  
  id_dpt int4 NOT NULL,  
  nom_departement text NULL,  
  id_region text NOT NULL,  
  CONSTRAINT departement_pkey PRIMARY KEY (id_dpt),  
  CONSTRAINT departement_id_region_fkey FOREIGN KEY (id_region) REFERENCES region(id_region)  
);
```

L'insertion des données devra suivre un certain ordre du fait des contraintes mises en place:

```
# Insertion des données, ordre d'insertion à respecter selon Fk  
Insertion_Data_Region()  
Insertion_Data_Departement()  
Insertion_Data_Artiste()  
Insertion_Data_Domaine()  
Insertion_Data_Inventaire()  
Insertion_Data_Musee()  
Insertion_Data_Oeuvre()  
Insertion_Data_Oeuvre_Artiste()  
Insertion_Data_Oeuvre_Domaine()  
Insertion_Data_Crea()  
Insertion_Data_Maj()
```

7 - Préparation des données

Cette phase de nettoyage a été la plus longue et la plus complexe du projet. Je suis face à des données saisies par différentes personnes, différents musées. Je rencontre une grande hétérogénéité des données.

Un nettoyage en amont minutieux est essentiel avant d'intégrer les données afin d'avoir des résultats d'analyses les plus fiables possibles.

Tout au long de ce nettoyage je me suis questionnée à savoir si je n'allais pas trop loin, espérant préserver au maximum l'intégrité des données.

Pour toute transformation majeure des données d'une colonne, je l'ai dupliquée afin de conserver son origine me permettant ainsi de comparer les modifications apportées avec les informations sources.

Je n'avais pas encore été confrontée à la manipulation de données textuelles.
C'est maintenant chose faite, j'ai beaucoup appris.

Mes compagnons ont été les bibliothèques Python : Pandas et Numpy, et Google !
Un grand merci à eux.

Notre enjeu premier : la chasse aux doublons masqués.

Les ressources principales pour mener à bien cette mission :

[Manipulation des données avec Pandas](#)

[Guide des expressions régulières — Documentation Python 3.9.2](#)

[Filtrer les données manquantes \(NaN, NULL\) d'une DataFrame avec Pandas ?](#)

[5 Must-Know Pandas Operations on Strings | by Soner Yıldırım](#)

[Splitting Columns With Pandas](#)

A suivre, les grandes lignes de notre épopée programmistique :

Phase 1 : actions menées suite aux premiers constats :

- Homogénéisation de l'ensemble des enregistrements en remplaçant tous les “;” par une “,” avec la méthode .replace car certaines cellules contiennent des “;” d'autres des “,” . J'ai également supprimé les tirés, car ils n'étaient pas saisis à enregistrement pour le même artiste.

Auteur	Date acquisition	Date de dépôt	Découverte / collecte	Dénomination	Lieu de dépôt	Mesures	...	Région	Sujet représenté	Source représentation	Statut juridique
DUPLESSIS Joseph Siffred	1886 ; 1897 Entrée matérielle	NaN	NaN	tableau	NaN	47 H ; 39 L	...	Hauts- de- France	portrait (Angiviller Charles comte d', homme, ...	NaN	propriété privée Personne morale;Interdiction ...

- Formatage en majuscules, minuscules ou première lettre en majuscule selon les colonnes avec les méthodes `.upper`, `.lower`, `.capitalize`
- Formatage des caractères non encodés UTF8. Malgré la lecture du csv en encodage UTF8, certains caractères ne sont pas traduits.

'ÃDOUARD Albert-Jules',
"ÃLOUIS Jean-Pierre-Henri, LEFÃVRE Robert (d'aprÃs)"]],

Première solution trouvée, lister les caractères spéciaux et les remplacer par le bon caractère :

```
sorted_df = sorted_df.replace(to_replace=("Ã©", "Ã¬", "Ã¯", "Ã", "Ã´",  
"Å§", "Äª", "Ä¹", "Ä¼", "Ä«", "Ä“", "à§", "äª", "ä¼", "ä^"),  
value=("é", "ï", "è", "à", "ô", "ç", "ê", "ù", "ü", "ë", "œ", "ç",  
"ê", "É", "ê"), regex=True)
```

```
'APPIANI Andrea', ..., "anonyme;ZIAM Félix (d'après)
'ÉDOUARD Albert-Jules',
"ÉLOUIS Jean-Pierre-Henri;LEFÈVRE Robert (d'après)"]
```

- Afin d'écartier les problèmes d'accents, majuscules omises et de m'assurer une suppression des doublons optimale, j'ai décidé de supprimer tout accent des colonnes Auteur, Titres, Ecole-Pays avec la méthode `.normalize()` et `encode('ascii')`, je finalise en appliquant le format majuscules.

DUPLESSIS JOSEPH SIFFRED', 'AITA DE LA PENUELA MATH
APPIANI ANDREA', ..., "ANONYME;ZIEM FELIX (D'APRES)
EDOUARD ALBERT-JULES',
'ELOUIS JEAN-PIERRE-HENRI;LEFEVRE ROBERT (D'APRES)"

Phase 2 : Problématiques de la table Artiste :

Comme évoqué précédemment, une œuvre peut avoir été réalisée par plusieurs artistes. De même qu'un artiste peut avoir un nom et un surnom. Une œuvre peut être peinte par un artiste mais cette œuvre sera attribuée à un autre artiste etc etc. Ici encore, je suis confrontée aux disparités de saisie de chaque personne.

Exemple caractéristique :

```
anonyme
anonyme (peintre)
HENNER Jean-Jacques
CHARNAY Armand
PICASSO Pablo (dit);RUIZ-PICASSO Pablo

MALLO Maruja
AUTHOUART Daniel
GRIMAUX Louis
PIJNACKER Adam
PIERO DI LORENZO (attribué);PIERO DI COSIMO (dit, attribué)
```

Mon idée est donc de supprimer les informations entre parenthèses et de splitter chaque élément après une virgule.

Dans un premier temps, le nez dans les données et le détail, à la recherche de solution, j'ai fait une liste de chaque valeur à supprimer :

```
sorted_df.Auteur_1 = sorted_df.Auteur_1.replace(to_replace=(" , D'APRES", "D'APRES, ",
" ,D'APRES", "D'APRES ", "D'APRES", "ATTRIBUE, D'APRES", "D'APRES", "ATTRIBUE, ",
", ATTRIBUE", "ATTRIBUE A, ENTOURAGE DE", "ATTRIBUE A", "ATTRIBUE", "ECOLE DE, \?",
"\(\?\)", "ECOLE DE", "ECOLE, D'APRES", "ECOLE D'", " , \?", "\?, ECOLE", "ECOLE,
D'APRES", "ECOLE", " , PEINTRE", "PEINTRE A", "PEINTRE, ", "PEINTRE", " , ATELIER",
"ATELIER, AUTEUR", "ATELIER DE", "ATELIER", "AUTEUR", "IMITATION", "ENTOURAGE DE",
"ENTOURAGE", "EXECUTANT", "GENRE DE", "\?", "MANUFACTURE", "MANIERE DE", "MANIERE",
"SUIVEUR DE", "SUIVEUR", "SUITE DE", "SCULPTEUR", "Ecrivain", "GRAVEUR",
"LITHOGRAPHE", "COPISTE", "CALLIGRAPHE", "DECORATEUR", "ELEVÉ DE", "ELEVÉ",
"PASTELLISTE", "ENCADREUR", "FORGERON", "DESSINATEUR", "COLLABORATEUR",
"ILLUSTRATEUR", "POUR LE PAYSAGE", "POUR LES PERSONNAGES", "ATTRIBUTION", "EDITEUR",
"IMITATEUR DE", "ARTISAN", "CREATEUR", " DU MODELE", "FABRICANT", "EMAILLEUR",
"AQUARELLISTE", "PHOTOGRAPHE", "IMPRIMEUR", "INCERTAINE DE", "INCERTAINE",
"PATRONYME", "NEE", "DIT A", "INSPIRE PAR", "SUR FAIENCE", "DIT, ", " , DIT", "DIT A
", " , DIT", "DIT,,", "DIT,", "\ (DIT\)", "\ (DITE\)", " DITE", " DIT", "\ (\)", "\ (,
\)", "\ ( \)", "\ (, \)", "\ )", "\ (", value="", regex=True)
```

Cette méthode doit vous faire bondir ! ^^

Oui, elle n'est pas optimale, surtout que je fais état des données du domaine peintures, que se passera-t-il pour le nettoyage des données d'autres domaines artistiques ? De plus, j'ai certainement omis des mentions.

Je me suis donc penchée plus en détail sur les méthodes regex et voilà la solution :

```
sorted_df[["Auteur_1"]] = sorted_df.Auteur.replace(regex=True,  
to_replace=(r"\(. *?\)", r"\(*", r"\) *"), value=(r' ', r' ', r' '))
```

Toute valeur entre parenthèses est supprimée ainsi que les valeurs entre parenthèses dans les parenthèses. Cette méthode peut être appliquée pour chaque domaine artistique.

Ressource pour cette optimisation :

[Guide des expressions régulières — Documentation Python 3.9.2](#)

Tests effectués sur <https://regexr.com/>

Phase 3 : Split des valeurs Auteur et Domaine :

Pour cette étape, j'ai utilisé la méthode split à partir de la première virgule et création d'une nouvelle colonne pour chaque élément.

8 colonnes auteur :

```
'Auteur_1', 'Auteur1', 'Auteur2', 'Auteur3', 'Auteur4', 'Auteur5', 'Auteur6', 'Auteur7', 'Auteur8'
```

9 colonnes domaine :

```
'Domaine_1', 'Domaine1', 'Domaine2', 'Domaine3', 'Domaine4', 'Domaine5', 'Domaine6', 'Domaine7',  
'Domaine8', 'Domaine9'
```

Source d'inspiration : [Splitting Columns With Pandas](#)

Phase 4 : Suppression des doublons :

Pour la suppression des doublons, j'ai évalué que l'on pouvait considérer comme doubles les œuvres ayant en commun la référence, l'identifiant Museofile, l'auteur et le titre, je conserve la première occurrence suite au tri effectué au préalable, paramétré de manière à avoir les valeurs nulles en fin de tri. Ainsi je m'assure de supprimer les lignes ayant le moins d'informations.

Etat des doublons avant suppression : `.duplicated().sum()`

5.576 valeurs doubles sur les 75.431 enregistrements.

Total final après méthode `drop.duplicated()` selon les critères cités précédemment de 69.855 œuvres uniques.

Phase 5 : Préparations des données pour les tables secondaires :

Démarche table Artiste :

pour rappel, voici les champs de cette table :

public.artiste		
123	id_artiste	serial
ABC	nom_artiste	text
ABC	precision_auteur	text
ABC	ecole_pays	text

Comme évoqué précédemment, le jeu de données liste plusieurs artistes, la précision auteur et école-pays font référence au premier nom d'artiste.

Je vais donc devoir faire la liste de chaque artiste et attribuer les informations précision auteur et l'école à l'artiste concerné.

voici ma démarche :

- Création d'un premier dataframe pandas :
`[["Auteur1", "P1", "Ecole_pays_1", "P_Dates", "Id"]]`
- Tri des données, valeurs nulles en dernier
- Suppression des doublons nom auteur
- Création d'un second dataframe :
`[["Auteur2", "Auteur3", "Auteur4", "Auteur5", "Auteur6", "Auteur7", "Auteur8", "Id"]]`
- Afin de récupérer chaque nom de chaque colonne, les lister, supprimer les doublons, je suis passée par un pivot avec la méthode `wide_to_long`

Ressource :

https://pandas.pydata.org/docs/reference/api/pandas.wide_to_long.html

`pd.wide_to_long(auteurs2, stubnames='Auteur', i='Id', j='N_Auteur')`

N_Auteur liste le chiffre de chaque colonne rattaché à stubnames Auteur.

		Auteur	
	Id	N_Auteur	
	13564	2	A
	13574	2	A

- Enfin j'utilise la méthode `pd.concat()` afin de regrouper les 2 dataframes, tri des colonnes, suppression des doublons en gardant la première occurrence.
- Enregistrement des données sous fichier Csv, celui-ci servira à l'insertion des données de la table Artiste

8 - Sauvegarde des données :

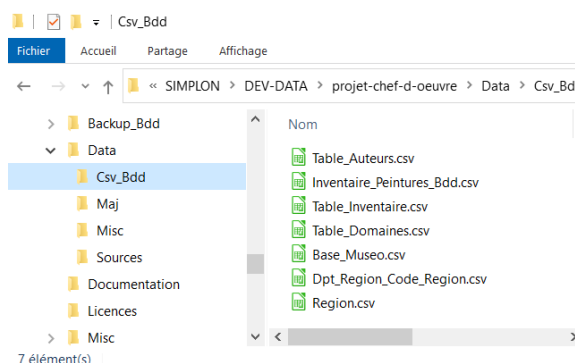
- Fichiers Csv :

Une sauvegarde des fichiers est effectuée à chaque étape du processus.

Les Csv exportés sont enregistrés sous Data/Sources.

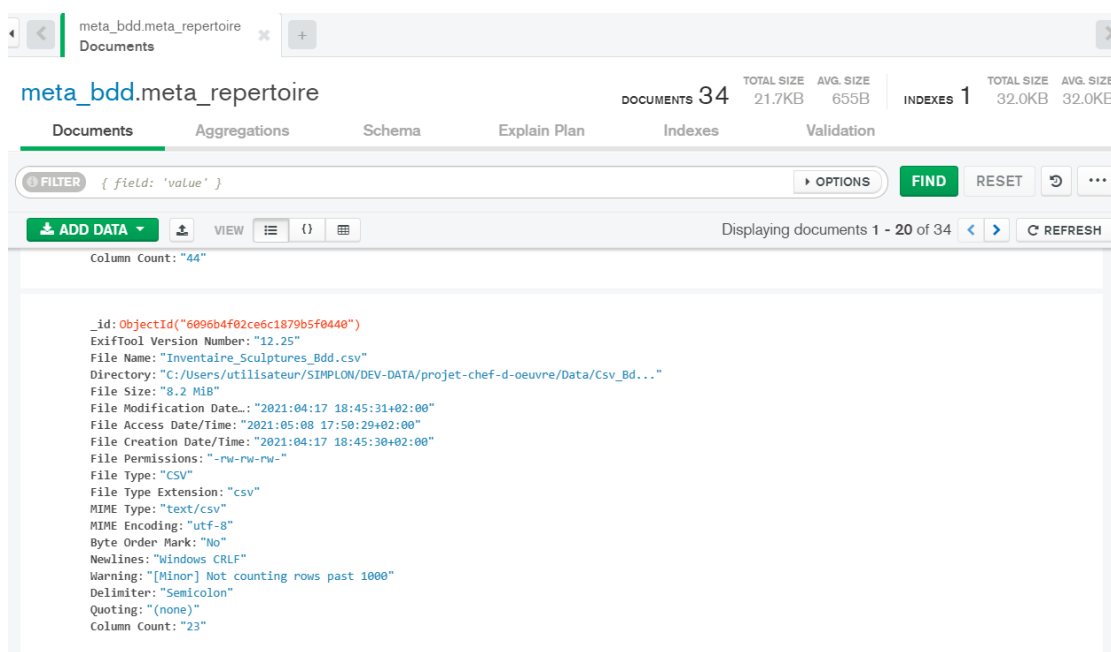
Les modifications apportées aux fichiers sources sont enregistrés sous Data/Maj.

Les fichiers finaux utilisés pour l'insertion des données sont enregistrés sous Data/Csv_Bdd.



- Métadonnées :

Le répertoire des métadonnées est généré sous script python et hébergé par MongoDB, SGBD non relationnel. Les métadonnées de chaque Csv importé, modifié sont répertoriées sous la collection meta_bdd.meta_repertoire.



- Un backup de sauvegarde de la base données est programmé chaque jour à 16h.
- Les éléments essentiels du projet sont poussés dans un répertoire Gitlab.

IV - INTERROGER LA BASE DE DONNÉES

1 - Requêtes SQL

J'arrive maintenant à l'étape tant attendue, interroger ma base de données. Après quelques tests satisfaisants, j'ai mis en place 2 vues SQL de jointures des tables Many to Many artistes/œuvres et catégorie/œuvres.

Je pourrai les utiliser afin de simplifier et alléger mes requêtes.

Mes ressources : [Cours de bases de données -- Modèles et langages](#)
[Cours et Tutoriels sur le Langage SQL](#)

Exemple : View v_Oeuvres_Artistes :

```
# Création vue Oeuvres avec infos Artistes et Titres d'oeuvres
creation_vue_Oeuvres_Artistes = """
CREATE VIEW v_Oeuvres_Artistes (
    Nom_Artiste,
    Id_Oeuvre,
    Titre_Oeuvre,
    Denomination_Oeuvre,
    Materieux_Techniques_Oeuvre,
    Sujet_Represente,
    Periode_Oeuvre,
    Id_Museo
)
AS
SELECT a.nom_artiste,
       o.id_oeuvre,
       o.titre_oeuvre,
       o.denomination,
       o.materiaux_techniques,
       o.sujet_represente,
       o.periode_creation,
       o.id_museo
FROM oeuvre o, oeuvre_artiste oa, artiste a

WHERE o.id_oeuvre = oa.id_oeuvre
AND oa.id_artiste = a.id_artiste;
"""
```

Test de la vue, recherche des oeuvres de l'artiste Rubens :

```
artistes_oeuvres = pd.read_sql_query("SELECT * FROM v_Oeuvres_Artistes WHERE nom_artiste LIKE '%RUBENS%' ORDER BY id_museo;", conn)
```

```
artistes_oeuvres.head()
```

	nom_artiste	id_oeuvre	titre_oeuvre	denomination_oeuvre	materieux_techniques_oeuvre	sujet_represente	periode_oeuvre	id_museo
0	RUBENS PETRUS PAULUS	9335	LE CHRIST TRIOMPHANT DE LA MORT ET DU PECHE, S...	tableau	peinture à l'huile, bois	figure biblique (Salvator Mundi, Christ, en pi...	1er quart 17e siècle	M0019
1	RUBENS PETRUS PAULUS	9334	SAINT FRANCOIS D'ASSISE	tableau	peinture à l'huile, toile	figure (saint François d'Assise, en pied, mouton)	1ère moitié 17e siècle	M0019
2	RUBENS PETRUS PAULUS	9218	SAINTE FAMILLE AVEC SAINT ELISABETH ET SAINT ...	tableau	peinture à l'huile, bois	scène biblique (Sainte Famille, Vierge, Enfant...	17e siècle	M0019
3	RUBENS PETRUS PAULUS	9215	LA VISITATION	tableau	peinture à l'huile, bois	scène biblique (Visitation, sainte Elisabeth, ...	1er quart 17e siècle	M0019

Je peux désormais pousser l'analyse et concevoir les requêtes répondant aux besoins du projet. A suivre, quelques exemples de résultats de requêtes:

Liste des artistes exposés par ville, le nom des musées exposants et nombre d'œuvres exposées :

Par exemple à Nantes :

(1069, 4)

	nom_artiste	nb_oeuvres	nom_musee	ville
0	ANONYME	530	musée des Beaux-Arts	Nantes
1	GORIN JEAN	54	musée des Beaux-Arts	Nantes
2	GORIN JEAN ALBERT	54	musée des Beaux-Arts	Nantes
3	LESAGE PIERRE ALEXIS	51	musée des Beaux-Arts	Nantes
4	SARKIS	42	musée des Beaux-Arts	Nantes

Recherche par nom d'artiste, lieux d'expositions et nombre d'œuvres :

Ici Léonard de Vinci :

(26, 4)

	nom_artiste	nb_oeuvres	nom_musee	ville
0	LEONARDO DI SER PIERO DA VINCI	10	musée du Louvre	Paris
1	VINCI LEONARD DE	8	musée du Louvre	Paris
2	LEONARD DE VINCI	5	musée Ingres	Montauban
3	LEONARD DE VINCI	3	musée des Beaux-Arts	Strasbourg
4	LEONARD DE VINCI	2	musée des Beaux-Arts	Nantes

Nombre d'œuvres exposées par un artiste par ville, nombre de musées par ville, pourcentage d'œuvres exposées par ville sur le total d'œuvres exposées de cet artiste en France.

Ici, Pierre Auguste Renoir :

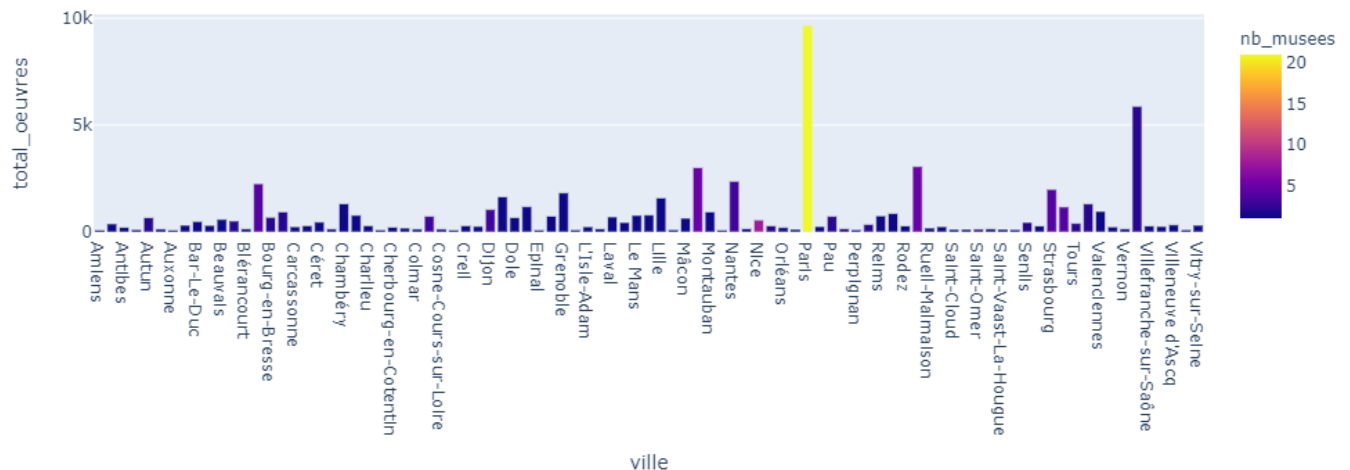
(13, 5)

	ville	nom_artiste	nb_musees	total_oeuvres	part_oeuvres
0	Paris	RENOIR PIERRE AUGUSTE	4	90	79.65 %
1	Bordeaux	RENOIR PIERRE AUGUSTE	1	6	5.31 %
2	Colmar	RENOIR PIERRE AUGUSTE	1	1	0.88 %
3	Dieppe	RENOIR PIERRE AUGUSTE	1	1	0.88 %
4	Grenoble	RENOIR PIERRE AUGUSTE	1	1	0.88 %
5	Le Havre	RENOIR PIERRE AUGUSTE	1	6	5.31 %
6	Lille	RENOIR PIERRE AUGUSTE	1	1	0.88 %
7	Nantes	RENOIR PIERRE AUGUSTE	1	1	0.88 %
8	Rouen	RENOIR PIERRE AUGUSTE	1	2	1.77 %
9	Soissons	RENOIR PIERRE AUGUSTE	1	1	0.88 %
10	Strasbourg	RENOIR PIERRE AUGUSTE	1	1	0.88 %
11	Autun	RENOIR PIERRE AUGUSTE	1	1	0.88 %
12	Versailles	RENOIR PIERRE AUGUSTE	1	1	0.88 %

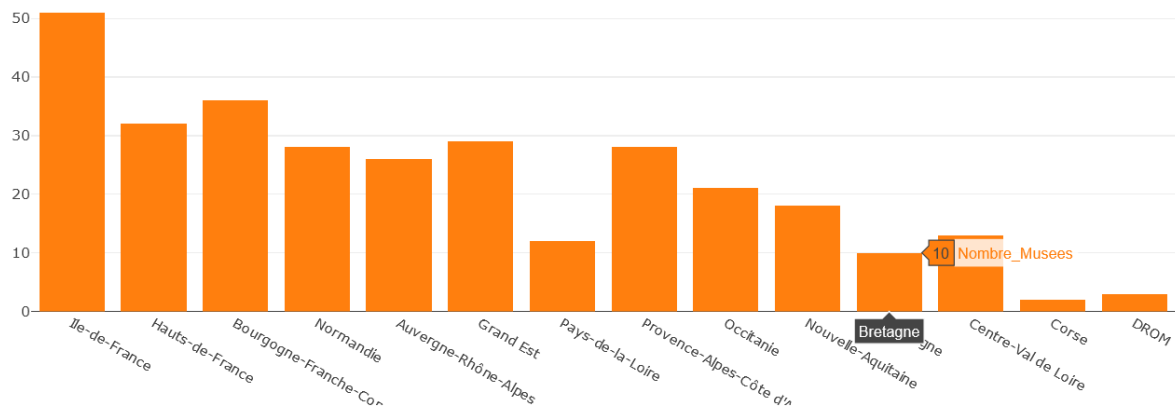
2 - Analyse des données sous représentations graphiques :

Ressources : [Folium Marker Clusters](#) / [Bar Charts](#) | [Python](#)

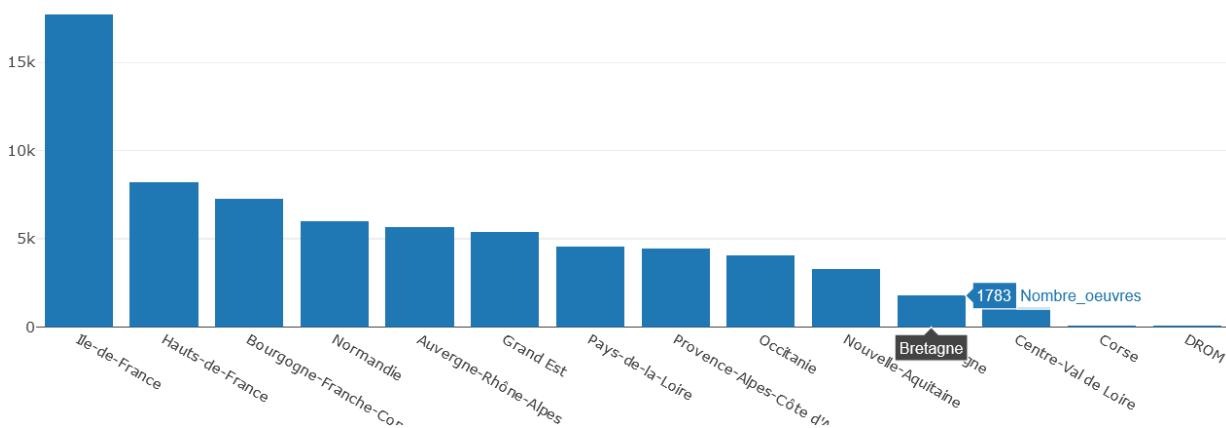
Les villes exposant plus de 100 œuvres du domaine artistique Peintures. Nous pouvons constater que Paris est en première position : 21 musées / 9.680 œuvres suivi de Versailles avec ses 5.888 œuvres.



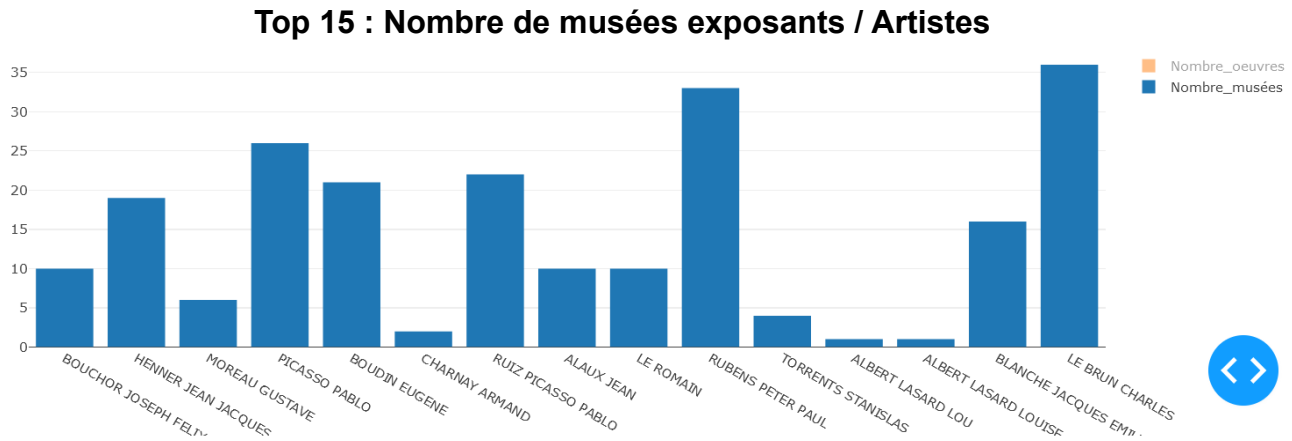
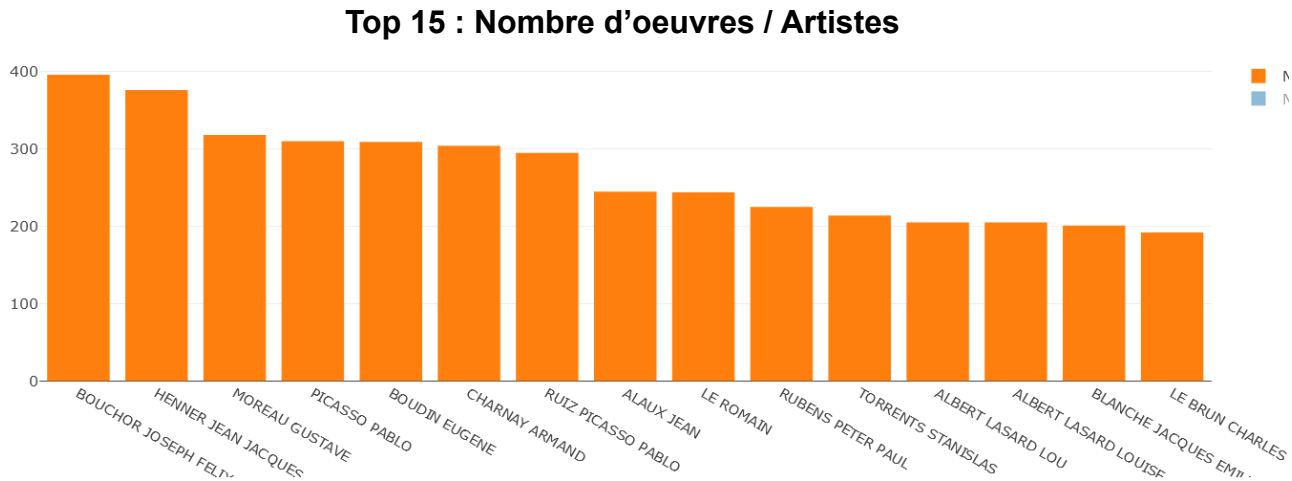
Nombre de musées exposants, domaine artistique peinture / Régions



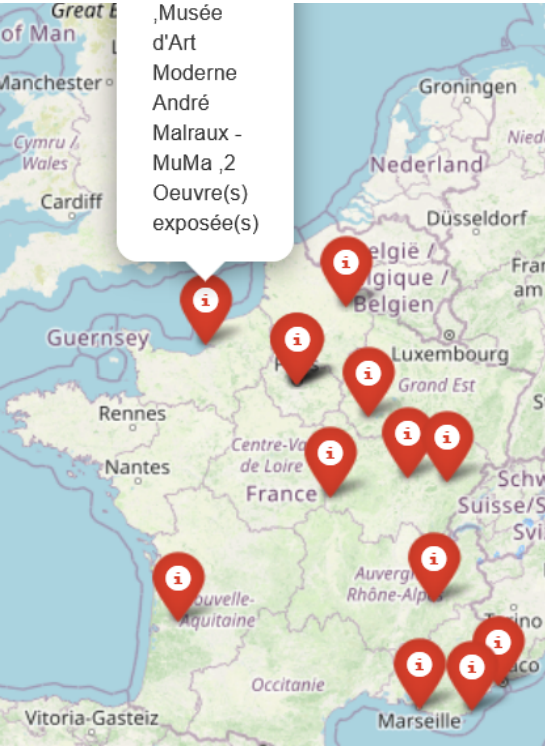
Nombre d'oeuvres, domaine artistique peinture / Régions



Top 15 des peintres exposés :



Géolocalisation des musées exposants les peintures de Matisse :



V - CONCLUSION

A ce jour, la base de données me semble opérationnelle pour concevoir des séjours sur la route des peintres.

En revanche, suite à ce que j'ai pu constater au cours du développement, les objectifs que je me suis donnée, cette version peut être optimisée sur plusieurs points. Voici les axes de réflexion :

- Alimenter la base de données avec les autres domaines Artistiques.
- Compléter les informations type : période de création / Epoque : les résultats sont biaisés par le peu d'informations saisies. J'aimerais les compléter par un scraping.
- Remise en question : faut-il garder les informations entre parenthèses des auteurs? Concernant les résultats des œuvres de Renoir exposées à Paris, j'obtiens un total de 97 œuvres.

La plateforme POP ressort ce résultat pour les mêmes critères :

[Lien de la recherche sur POP](#)

Auteur

renoir

- ☐ RENOIR Pierre Auguste (83)
- ☐ RENOIR Auguste (dit) (7)
- ☐ RENOIR Pierre-Auguste (7)

vs Bdd recherche 'like' RENOIR à Paris :

(2, 5)

	ville	nom_artiste	nb_musees	total_oeuvres	part_oeuvres
0	Paris	RENOIR PIERRE AUGUSTE	4	90	70.87 %
1	Paris	RENOIR AUGUSTE	1	7	5.51 %

Devrais-je garder la mention (dit) pour cibler les faux doublons et les exclure de mes requêtes ?

- Mettre en place un script permettant de calculer la distance entre chaque musée.
- Mettre en place crontab pour le backup plutôt que schedule.
- Développer une interface IHM.
- Optimiser le script.

Enfin, pour aller plus loin, j'aimerais contacter une personne de l'équipe de Data gouv et échanger avec elle sur mon projet. J'aimerais également avoir plus d'informations sur la manière dont les inventaires sont réalisés. L'inventaire fait-il état des œuvres réellement exposées ?

Quelle est la fréquence de la mise à jour des bases Joconde et Museofile ?

Je vais également passer au musée des beaux arts de Nantes et en savoir un peu plus !
Ce sera l'occasion de faire une petite visite des œuvres exposées !