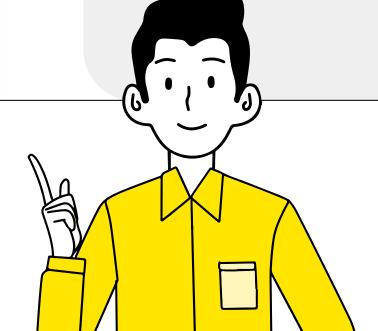
# Concours sur la Smart City

Parcours Al Engineer



#### Notre mission:

# Optimiser des tournées pour l'entretien des arbres de la ville .



# Les étapes d'analyse

Partie 1

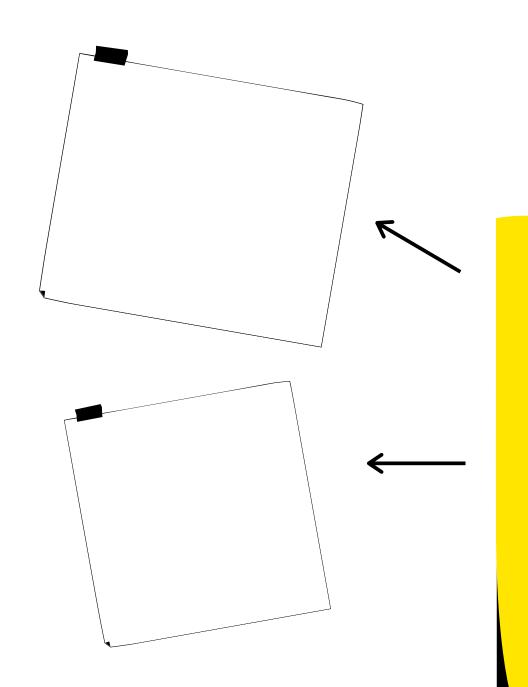
Exploration et Documentation

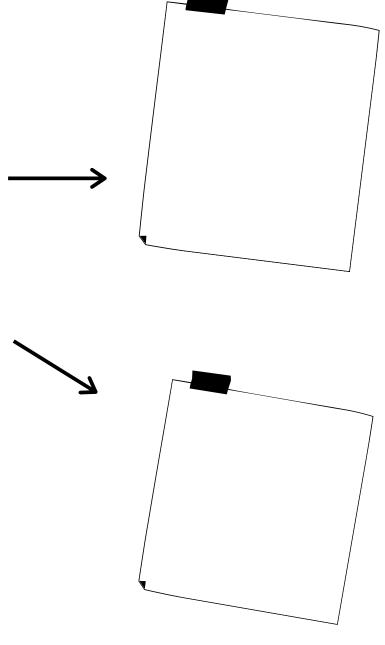
Partie 2

- Approfondissement et étude
- Mise en place sur un arrondissement

Partie 3

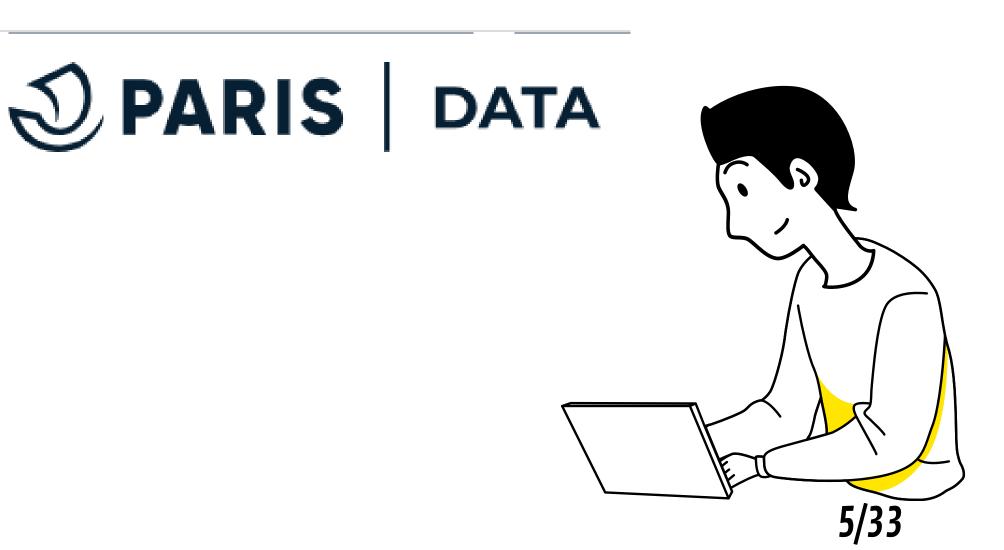
- Géolocalisation
- Pour aller plus loin





# Source externe

- Documentation sur le site de la ville de Paris
- Youtube et reportage
- Internet



# Jeu de données

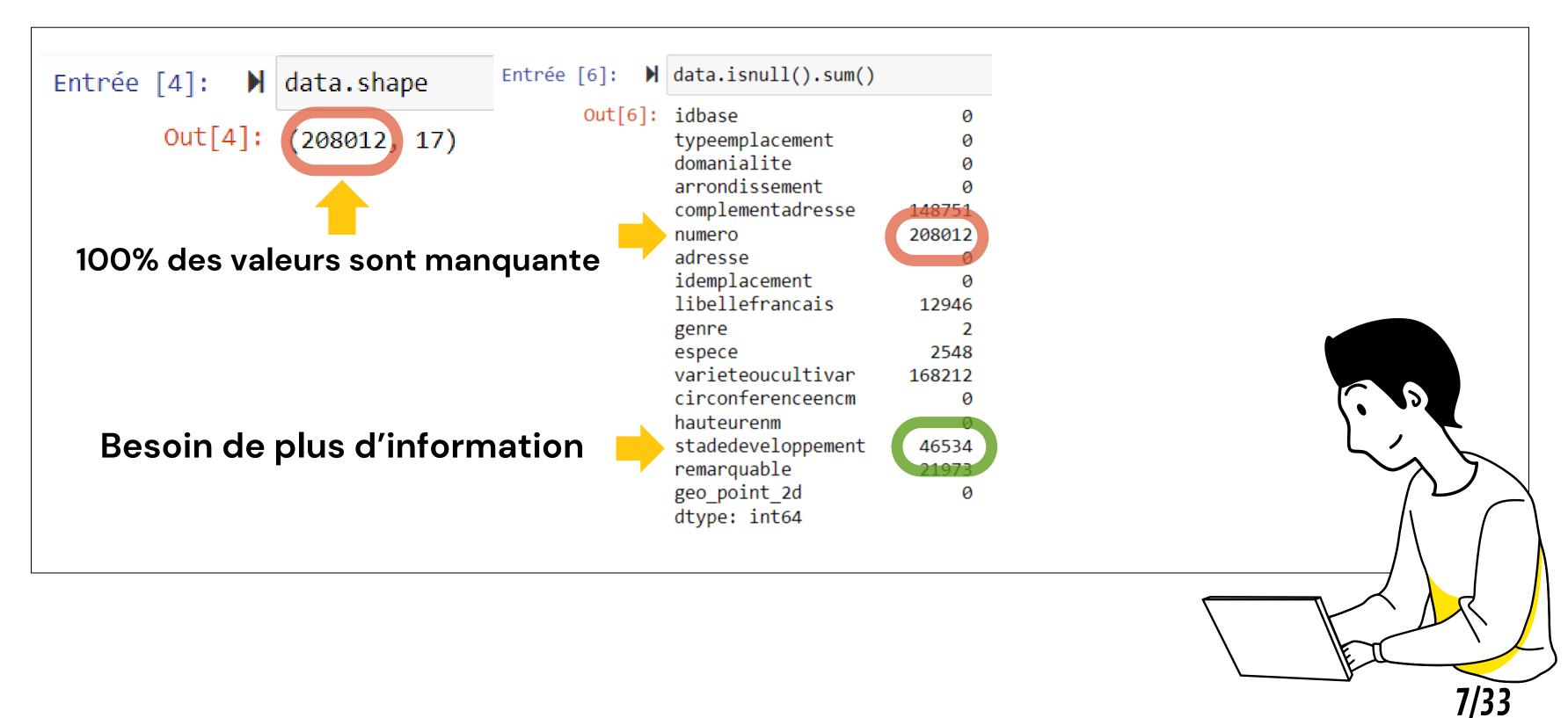
- Stade de développement.
- Domanialité.
- Hauteur, geopoint, arrondissement.

```
data.shape
(208012, 17)
```

colonnes = data.columns.tolist()
print(colonnes)

```
id_base, type_emplacement,
domanialite, arrondissement,
complement_adresse, numero,
adresse, id_emplacement,
libelle_francais, genre, espece,
varieteoucultivar,
circonference_en_cm, hauteur_en_m,
stade_developpement, remarquable,
geo_point_2d
```

Observation



Corrélation arbres matures et remarquables

stadedeveloppement
Adulte 78487

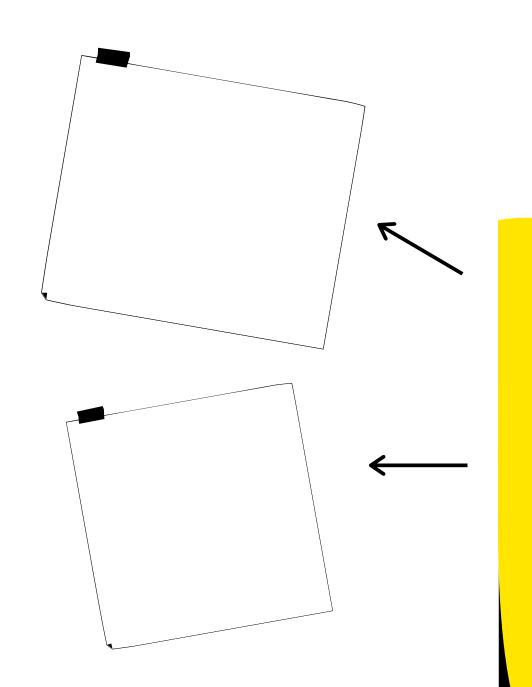
Jeune (arbre) 36966

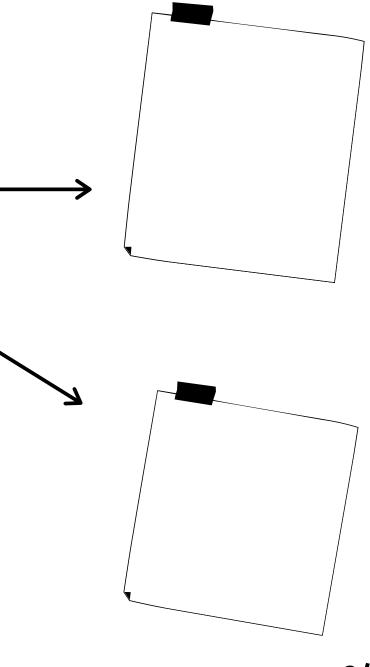
Jeune (arbre)Adulte Mature 7895

188

Lien?

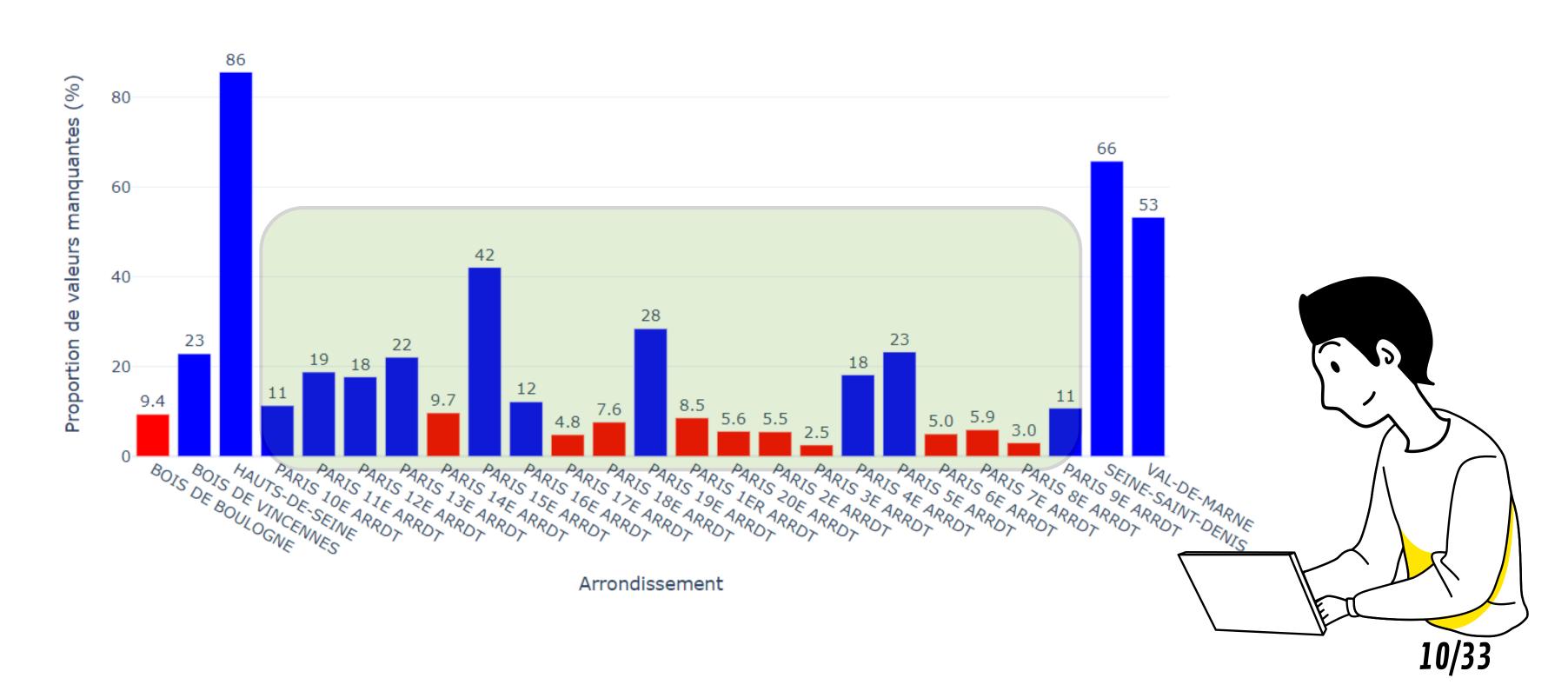






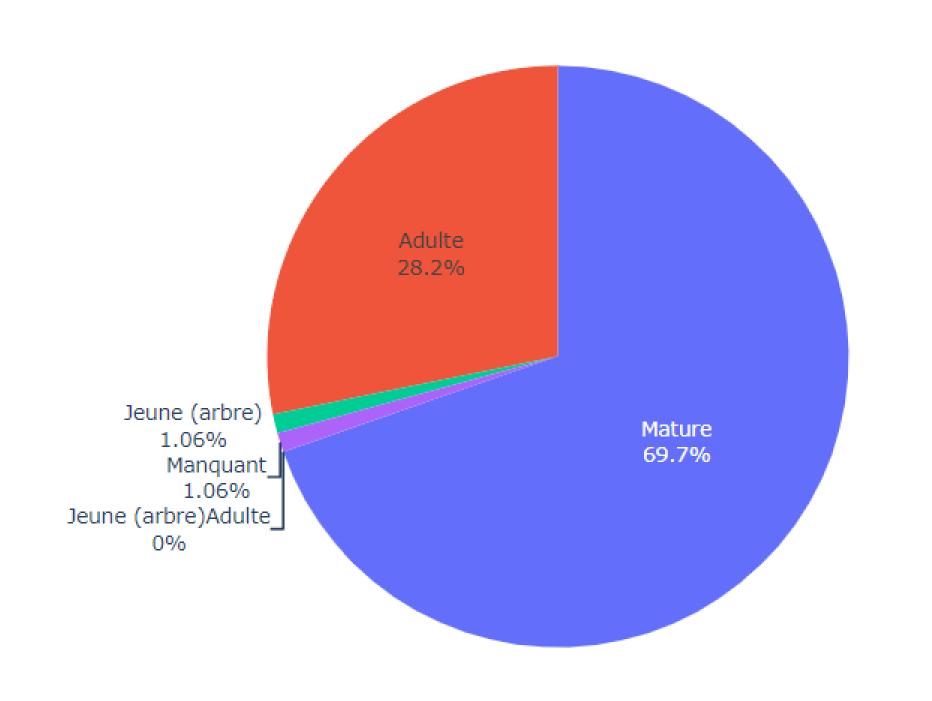
Observation des données manquantes du stade de développement

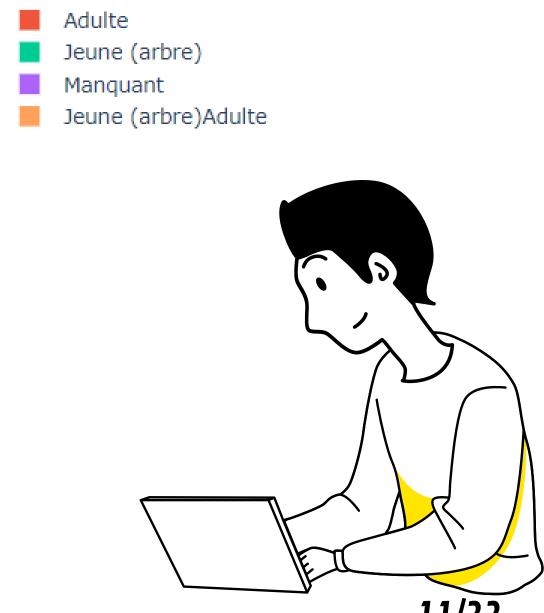
Proportion de valeurs manquantes de stade de développement par arrondissement



Corrélation arbres matures et remarquables

Répartition des arbres remarquables par stade de développement





Stade de developpement

Mature

Nettoyage - Clarté

#### Renomme les domanialités pour plus de compréhension

domanialite	
DASES	18
DAC	67
DFPE	1416
DJS	4753
PERIPHERIQUE	5219
DASCO	7307
CIMETIERE	31875
Jardin	50688
Alignement	106669



domanialite	
Action_sociale	18
Équipements_culturels	67
Creches	1416
Equipements_sportifs	4753
PERIPHERIQUE	5219
Ecoles	7307
CIMETIERE	31875
Jardin	50688
Alignement	106669

• On supprime la colonne 'numéro' sans donnée.

#Suppression de la colonne numéro à 100% de valeurs manquantes data = data.drop('numero', axis=1)



Nettoyage - Doublon

```
data.duplicated('idbase').sum()
1

1
20
5 Jeune (arbre)
```

circonferenceencm hauteurenm stadedeveloppement rer

```
#Traitement du doublon :
#Correction de la valeur
index_to_update = 73576
new_value = 'Jeune (arbre)Adulte'
data.loc[index_to_update, 'stadedeveloppement'] = new_value

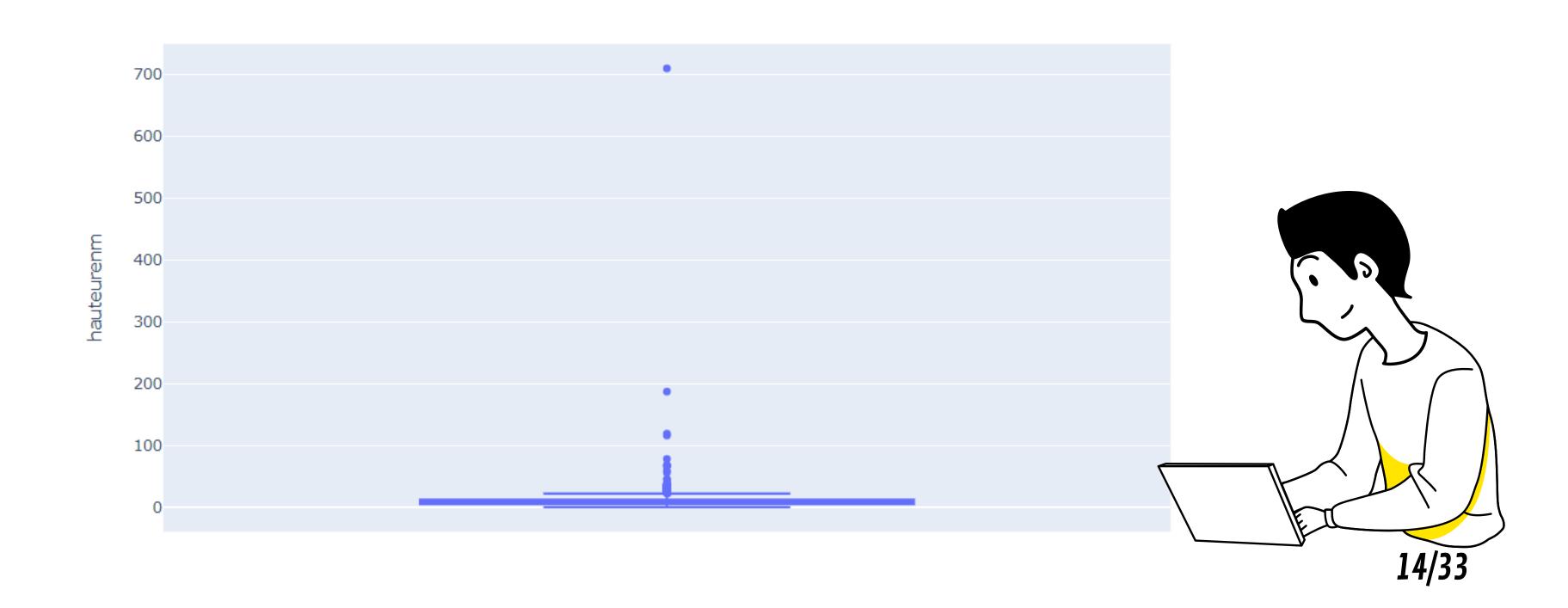
#suppression du doublon 125873
index_to_remove = 125873
data = data.drop(index_to_remove)

# Réinitialisez les index après la suppression pour éviter des indices manquants
data = data.reset_index(drop=True)
```



Nettoyage - Valeurs aberrantes

```
fig = px.box(data, y='hauteurenm')
fig.show()
```



Nettoyage - Valeurs aberrantes

25073

1195

```
data.groupby('hauteurenm')['idbase'].count()
#On va considérer que les arbres de 0m sont des valeurs abérantes
#On va considérer que les arbres entre 35 et 60m sont atypiques
#On va considérer que les arbres au dessus de 60m sont abérantes
hauteurenm
```

# 68 1 78 1 116 1 119 1 187 1 710 1

67

```
condition = (data['hauteurenm'] == 0) & (data['stadedeveloppement'].isin(['Mature', 'Adulte', 'Jeune (arbre)', 'Jeune (arbre)
```

Il y a 1198 arbres de 0m de haut

Il y a 9 arbres de plus de 60m de haut



Nettoyage - Valeurs aberrantes

```
Om:

Adulte 684 Adulte 5

Jeune (arbre)Adulte 289 Jeune (arbre)Adulte 3

Jeune (arbre) 213 Mature 1

Name: stadedeveloppement, dtype: int64
```

```
Moyenne de 'hauteurenm' pour les individus 'Adulte' : 13
Moyenne de 'hauteurenm' pour les individus 'Jeune (arbre)' : 5
Moyenne de 'hauteurenm' pour les individus 'Jeune (arbre)Adulte' : 8
Moyenne de 'hauteurenm' pour les individus 'Mature' : 20
```

```
# Modifier les valeurs de 'hauteurenm' en fonction des conditions spécifiées
data.loc[(data['hauteurenm'] > 60) & (data['stadedeveloppement'] == 'Adulte'), 'hauteurenm'] = 13
data.loc[(data['hauteurenm'] > 60) & (data['stadedeveloppement'] == 'Jeune (arbre)Adulte'), 'hauteurenm'] = 8
data.loc[(data['hauteurenm'] > 60) & (data['stadedeveloppement'] == 'Mature'), 'hauteurenm'] = 20
```



Nettoyage - Contrôle texte

```
data.groupby('libellefrancais')
libellefrançais
Platane
                     42272
Marronnier
                     24157
Erable
                     18676
Tilleul
                     17413
Sophora
                     11517
Pin
                      4903
Micocoulier
                      4618
Frêne
                      4580
Cerisier à fleurs
                      4191
Chêne
                      4033
Name: idbase, dtype: int64
```

```
# Test de 'libellefrancais' pour verifier les fautes de frappes.
mot correct = 'Chêne'
# Filtrer les valeurs nulles dans la colonne 'libellefrancais'
donnees non nulles = [x for x in data['libellefrancais'] if x is not None]
# Trouver les correspondances potentielles avec un score de similarité minimale de 80
resultats = process.extract(mot correct, donnees non nulles, limit=20, scorer=fuzz.ratio)
# Afficher les résultats en excluant les correspondances parfaites (scores de 100)
for resultat in resultats:
    mot_potentiel, score = resultat
    if score < 100: # Exclure les scores de 100
        print(f'Mot potentiel: {mot potentiel}, Score de similarité: {score}')
```

Calcul de l'âge

#### Calcul de l'âge de l'arbre (Non-Utilisé)

```
#data.groupby('libellefrancais')['idbase'].count()

#data['Age_Arbre'] = (data['circonferenceencm'] / np.pi) * 2.25
#data['Age_Arbre'] = data['Age_Arbre'].astype(int)
#data[data['stadedeveloppement'] == 'Jeune (arbre)Adulte']

#data[data['stadedeveloppement'] == 'Jeune (arbre)']
```

#### Le multiplicateur pose problème

stadedeveloppement Adulte Jeune (arbre) Jeune (arbre)Adulte Mature



Création de l'indice de difficulté d'intervention

6.0

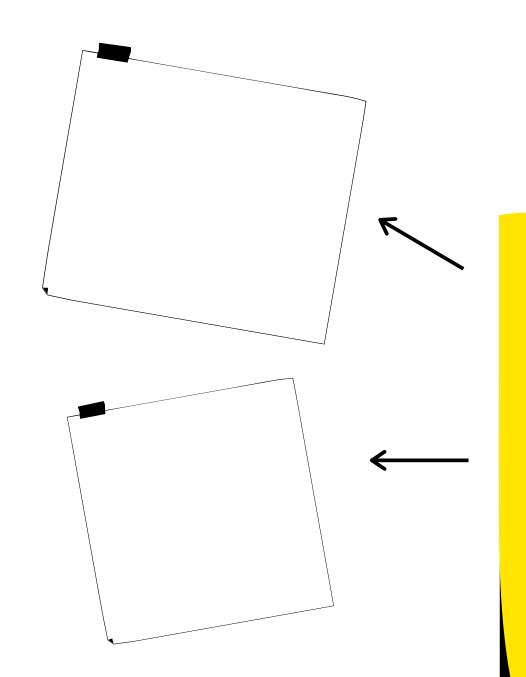
60371

Regroupement par taille

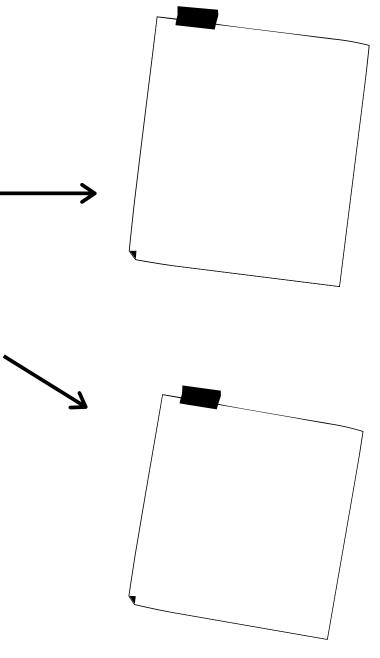
```
taille_group
1-3 3807
3-10 86621
>10 93708
```

 Création de l'indice

```
data.loc[:, 'Indice_Intervention'] = 0
  # Dictionnaires de correspondance des difficultés
  taille_difficulte = {'1-3': 0, '3-10': 1, '>10': 2}
  dominalite_difficulte = {'Jardin': 0, 'CIMETIERE': 1, 'PERIPHERIQUE': 3, 'Alignement':4,
                          'Creches':1.5, 'Ecoles':1.5, 'Equipements_sportifs': 0.5,
Indice_Intervention 'Équipements_culturels':3,'Action_sociale':2}
0.0
         12200
0.5
           222
         35014
1.0
1.5
          4405
2.0
         25687
2.5
          5748
3.0
         10009
3.5
          3101
4.0
          2914
5.0
         48340
```



# Partie 2: Mise en place sur un arrondissement.



# Partie 2: Mise en place sur un arrondissement.

Sélection de l'arrondissement

#### Total arbres par arrondissement | Pourcentage de val. manquante

LANTO TIT WIND!	0041
PARIS 8E ARRDT	7440
VAL-DE-MARNE	7577
PARIS 7E ARRDT	8718
PARIS 18E ARRDT	11049
PARIS 14E ARRDT	11632
PARIS 17E ARRDT	11730
BOIS DE VINCENNES	11816
SEINE-SAINT-DENIS	12126
PARIS 12E ARRDT	12763
PARIS 19E ARRDT	14874
PARIS 20E ARRDT	15656
PARIS 13E ARRDT	16944
PARIS 16E ARRDT	17084
PARIS 15E ARRDT	17490

PARIS 3E ARRDT	2.5
PARIS 8E ARRDT	3.0
PARIS 17E ARRDT	4.8
PARIS 6E ARRDT	5.0
PARIS 2E ARRDT	5.5
PARIS 20E ARRDT	5.6
PARIS 7E ARRDT	5.9
PARIS 18E ARRDT	7.6
PARIS 1ER ARRDT	8.5
BOIS DE BOULOGNE	9.4
PARIS 14E ARRDT	9.7
PARIS 9E ARRDT	10.7
PARIS 10E ARRDT	11.3
PARIS 16E ARRDT	12.1
DADTE 40E ADDDT	47 7



#### Partie 2: Mise en place sur un arrondissement.

Observation de tendance.

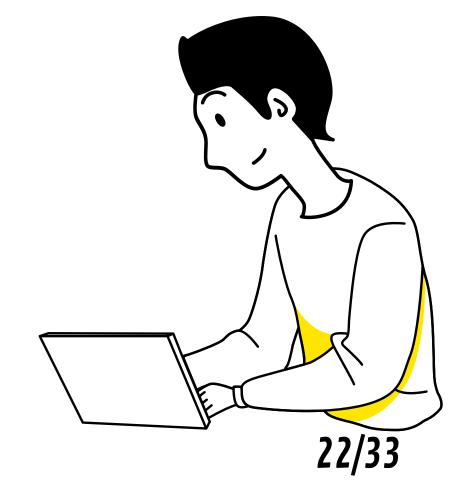
```
paris18E = data[data['arrondissement'] == 'PARIS 18E ARRDT']
#paris18E

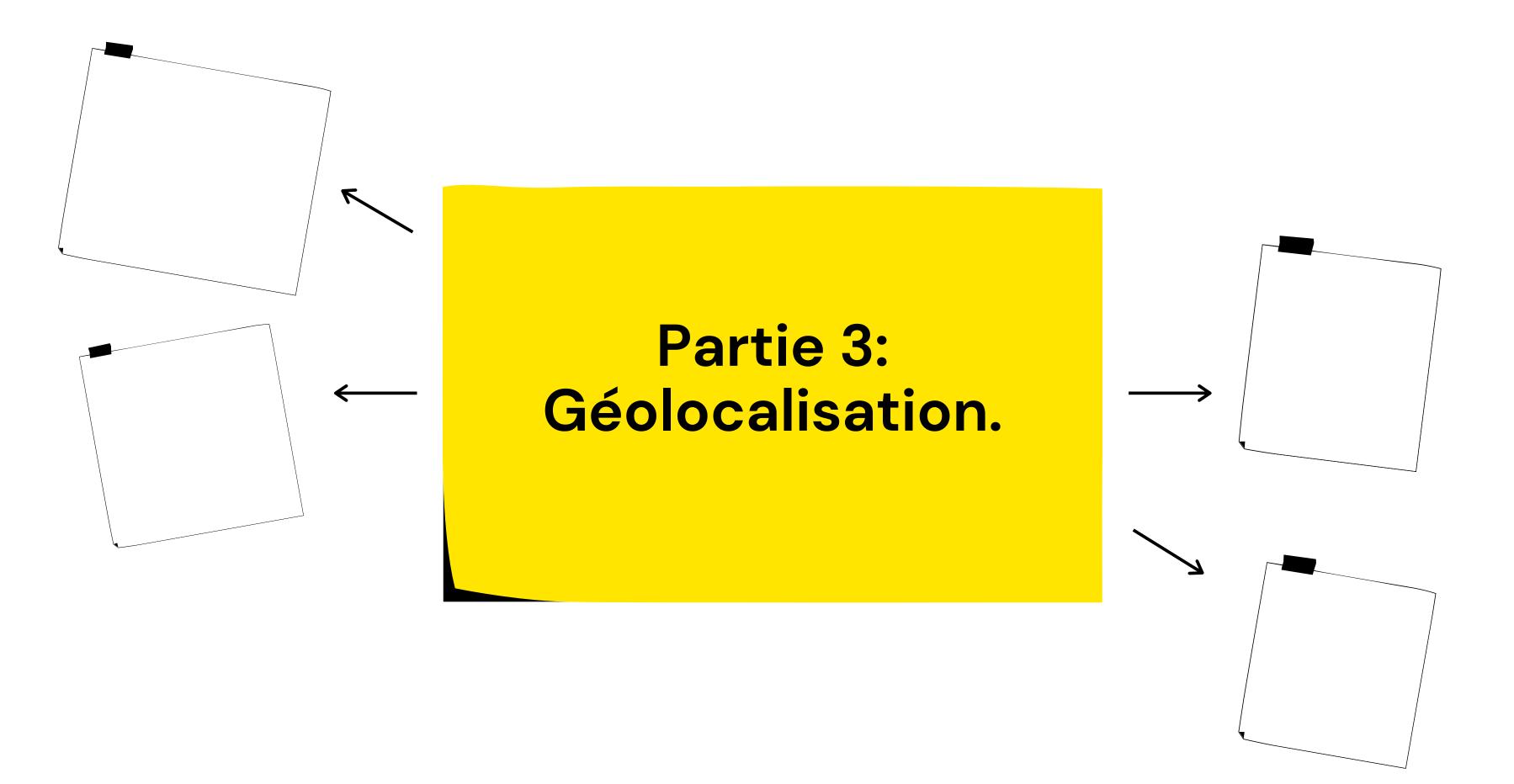
#paris18E.groupby('domanialite')['idbase'].count()

#paris18E.groupby('stadedeveloppement')['idbase'].count()

#paris18E.groupby('taille_group')['idbase'].count()

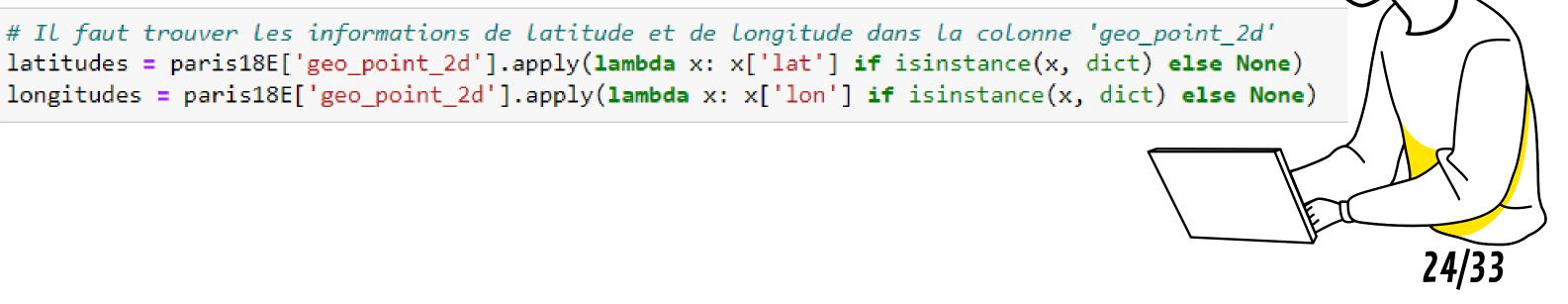
#paris18E.groupby('Indice_Intervention')['idbase'].count()
```





Mise en place

```
pd.set_option('display.max_colwidth', None)
print(paris18E['geo_point_2d'])
          {'lon': 2.366573402189964, 'lat': 48.900070506864395}
5
           {'lon': 2.359779075854935, 'lat': 48.90124746445273}
            {'lon': 2.341989385401331, 'lat': 48.8834186651702}
75
           {'lon': 2.351857471395496, 'lat': 48.89408076784284}
86
           {'lon': 2.361889977996846, 'lat': 48.89880380873941}
92
         {'lon': 2.353009455809968, 'lat': 48.890874285982335}
207984
         {'lon': 2.331328656497397, 'lat': 48.890992881008636}
208000
         {'lon': 2.338602926200885, 'lat': 48.89769647986777}
208002
         {'lon': 2.364643366786766, 'lat': 48.895079416006304}
208005
          {'lon': 2.333844112582832, 'lat': 48.900589681194795}
208007
Name: geo point 2d, Length: 11049, dtype: object
```



#### Nettoyages

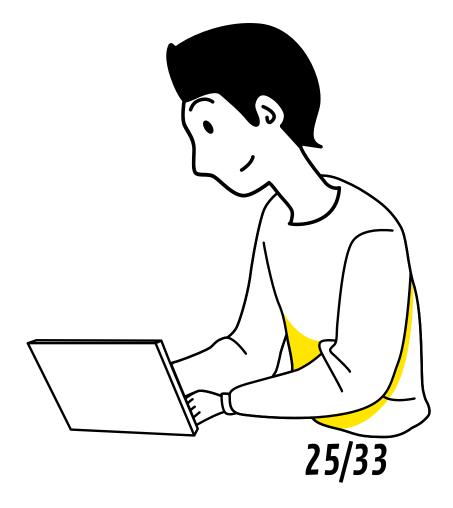
```
paris18E.duplicated('geo_point_2d').sum()
```

43

```
# Verification de valeurs 1
valeur_specifique = {'lon': 2.367349005766865, 'lat': 48.88849211186056}

# Identifier les doublons de la valeur spécifique dans paris18E
doublons_specifiques = paris18E[paris18E['geo_point_2d'] == valeur_specifique]

# Affichage des résultats
print("Doublons de la valeur spécifique :")
print(doublons_specifiques)
```



Nettoyages

Pourquoi 'geo\_point\_2d' et pas 'idemplacement' ?

Pourquoi sur paris18E et pas le jeu complet?



Nous allons créer trois cartes de géolocalisation, et voici comment les interpréter et les utiliser :

#### 1. Carte de Difficulté d'Intervention :

L'objectif de cette carte est de bien préparer l'intervention en amont dans le secteur. Elle vise à permettre à l'équipe d'intervention de disposer du matériel nécessaire en fonction de la difficulté de l'intervention sur l'arbre. Cela inclut l'équipement pour monter à l'arbre, ainsi que le matériel nécessaire pour sécuriser la zone et éventuellement adapter la circulation.

#### 2. Carte des Différents Stades de Développement :

Cette carte permet d'organiser e<mark>t d'anticiper la fréquence des interventions</mark>, ainsi que l'attention à accorder aux différents arbres du secteur, en fonction de leur stade de développement.

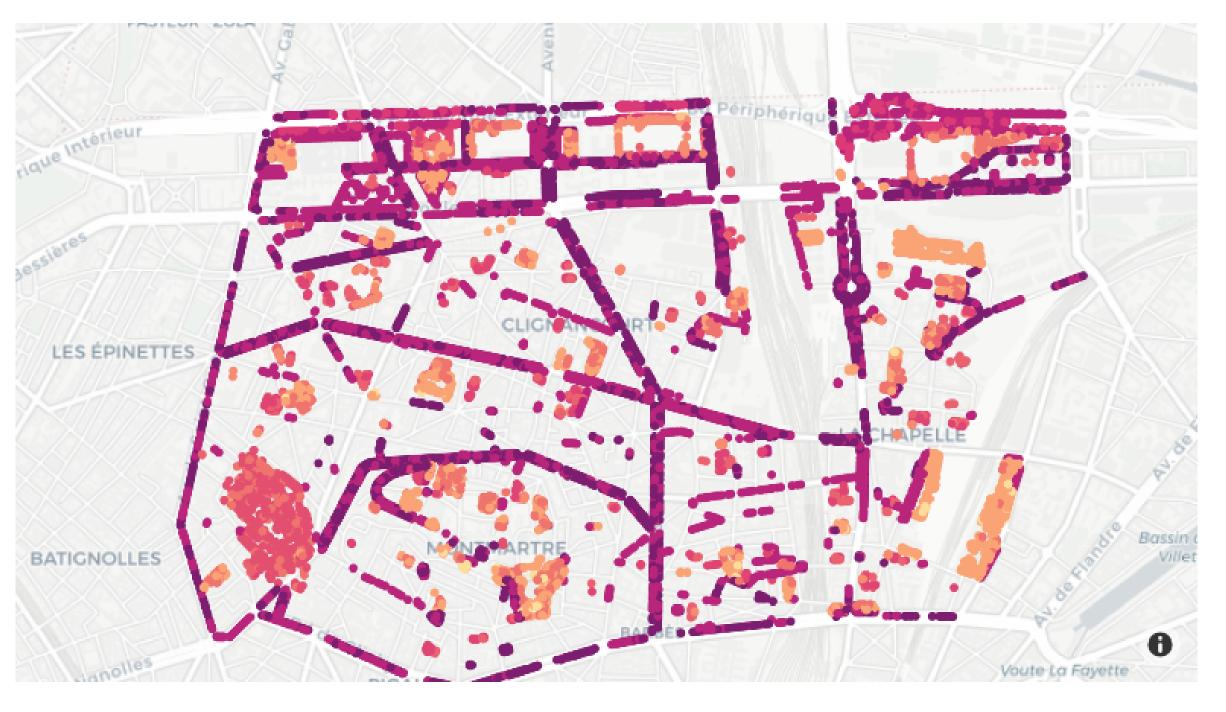
#### 3. Carte des Genres et Espèces :

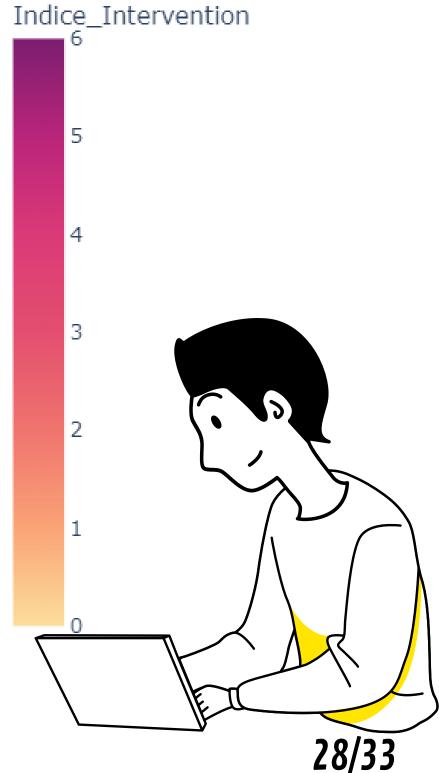
La carte des genres et espèces renseigne sur les risques potentiels de propagation des maladies. Elle aide à se préparer en observant de manière plus précise les risques éventuels du secteur d'intervention.

Ces cartes joueront un rôle important dans la planification efficace des interventions, en permettant d'adapter les ressources et l'approche en fonction des spécificités de chaque situation.

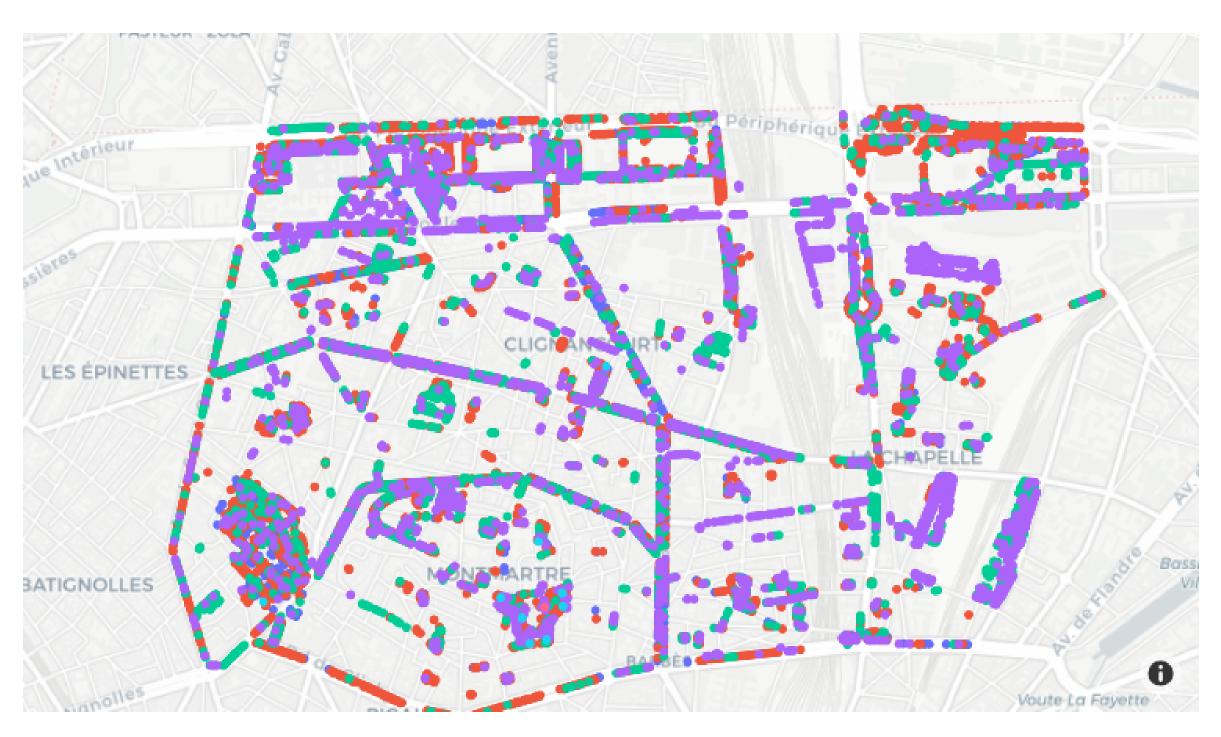


#### Carte de Difficulté d'Intervention





Carte des Différents Stades de Développement

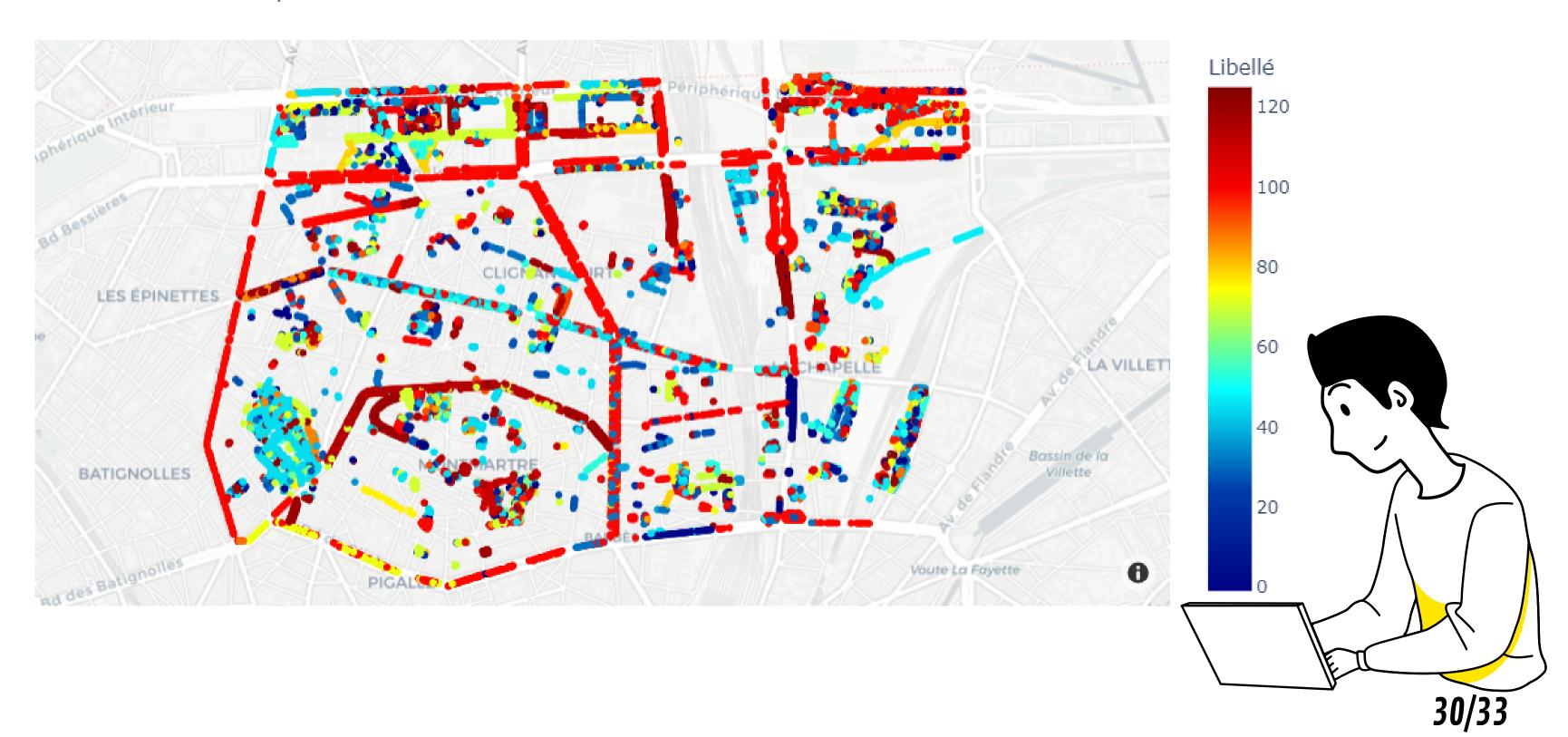


#### Stade de developpement

- Mature
- Adulte
- Jeune (arbre)Adulte
- Jeune (arbre)
- Mature (Remarquable)
- Adulte (Remarquable)



#### Carte des Genres et Espèces







# Direction le NoteBook!

