In [2]:
```
"""
Created on Thu Jan 30 17:46:18 2025

@author: Thierry ALLEM


TRAITEMENT DE LA BASE DE DONNES SYNOP INCLUANT DES DONNEES METEOROLOGIQUES

"""
```

Out[2]:
```
'\nCreated on Thu Jan 30 17:46:18 2025\n\n@author: Thierry ALLEM\n\n\nTRAITEMENT DE LA BASE DE DONNES SYNOP INCLUANT DES DONNEES METEOROLOGIQUES\n\n'
```

In [3]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import os
from tqdm import tqdm
import warnings
warnings.filterwarnings('ignore')
```

In [4]:
```
# Importation du fichier des relevés de températures
# Lecture du fichier CSV
df_meteo = pd.read_csv("donnees-synop-essentielles-omm.csv", sep=';', encoding='utf-8')
df_meteo.head()
```

Out[4]:

	ID OMM station	Date	Pression au niveau mer	Variation de pression en 3 heures	Type de tendance barométrique	Direction du vent moyen 10 mn	Vitesse du vent moyen 10 mn	Température	Point de rosée	Humidité	...	Altitude	communes (name)	communes (code)	EPCI (name)	EPCI (code)
0	7558	2010-01-05T10:00:00+01:00	100280.0	-50.0	5.0	260.0	1.5	275.75	275.75	100.0	...	712	Millau	12145	CC de Millau Grands Causses	241200567.0
1	61976	2010-01-05T10:00:00+01:00	100990.0	NaN	NaN	NaN	NaN	305.45	299.05	69.0	...	7	NaN	NaN	NaN	NaN
2	7027	2010-01-05T13:00:00+01:00	100720.0	-190.0	8.0	200.0	3.6	273.65	271.75	87.0	...	67	Carpiquet	14137	CU Caen la Mer	200065597.0
3	7110	2010-01-05T13:00:00+01:00	100750.0	-230.0	8.0	210.0	4.1	276.95	272.55	73.0	...	94	Guipavas	29075	Brest Métropole	242900314.0
4	7591	2010-01-05T13:00:00+01:00	NaN	NaN	NaN	NaN	NaN	274.45	269.05	67.0	...	871	Embrun	05046	CC Serre-Ponçon	200067742.0

5 rows × 82 columns

In [5]:
```
df_meteo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2557951 entries, 0 to 2557950
Data columns (total 82 columns):
 #   Column                                                Dtype
---  ------                                                -----
 0   ID OMM station                                        int64
 1   Date                                                  object
 2   Pression au niveau mer                                float64
 3   Variation de pression en 3 heures                     float64
 4   Type de tendance barométrique                         float64
 5   Direction du vent moyen 10 mn                         float64
 6   Vitesse du vent moyen 10 mn                           float64
 7   Température                                           float64
 8   Point de rosée                                        float64
 9   Humidité                                              float64
 10  Visibilité horizontale                                float64
 11  Temps présent                                         float64
 12  Temps passé 1                                         float64
 13  Temps passé 2                                         float64
 14  Nebulosité totale                                     float64
 15  Nébulosité  des nuages de l' étage inférieur          float64
 16  Hauteur de la base des nuages de l'étage inférieur    float64
 17  Type des nuages de l'étage inférieur                  float64
 18  Type des nuages de l'étage moyen                      float64
 19  Type des nuages de l'étage supérieur                  float64
 20  Pression station                                      float64
 21  Niveau barométrique                                   float64
 22  Géopotentiel                                          float64
 23  Variation de pression en 24 heures                    float64
 24  Température minimale sur 12 heures                    float64
 25  Température minimale sur 24 heures                    float64
 26  Température maximale sur 12 heures                    float64
 27  Température maximale sur 24 heures                    float64
 28  Température minimale du sol sur 12 heures             float64
 29  Méthode de mesure Température du thermomètre mouillé  float64
 30  Température du thermomètre mouillé                    float64
 31  Rafale sur les 10 dernières minutes                   float64
 32  Rafales sur une période                               float64
 33  Periode de mesure de la rafale                        float64
 34  Etat du sol                                           float64
 35  Hauteur totale de la couche de neige, glace, autre au sol  float64
 36  Hauteur de la neige fraîche                           float64
 37  Periode de mesure de la neige fraiche                 float64
 38  Précipitations dans la dernière heure                 float64
 39  Précipitations dans les 3 dernières heures            float64
 40  Précipitations dans les 6 dernières heures            float64
 41  Précipitations dans les 12 dernières heures           float64
 42  Précipitations dans les 24 dernières heures           float64
 43  Phénomène spécial 1                                   float64
 44  Phénomène spécial 2                                   float64
 45  Phénomène spécial 3                                   float64
```

```
 46  Phénomène spécial 4                                  float64
 47  Nébulosité couche nuageuse 1                         float64
 48  Type nuage 1                                         float64
 49  Hauteur de base 1                                    float64
 50  Nébulosité couche nuageuse 2                         float64
 51  Type nuage 2                                         float64
 52  Hauteur de base 2                                    float64
 53  Nébulosité couche nuageuse 3                         float64
 54  Type nuage 3                                         float64
 55  Hauteur de base 3                                    float64
 56  Nébulosité couche nuageuse 4                         float64
 57  Type nuage 4                                         float64
 58  Hauteur de base 4                                    float64
 59  Coordonnees                                          object
 60  Nom                                                  object
 61  Type de tendance barométrique.1                      object
 62  Temps passé 1.1                                      object
 63  Temps présent.1                                      object
 64  Température (°C)                                     float64
 65  Température minimale sur 12 heures (°C)              float64
 66  Température minimale sur 24 heures (°C)              float64
 67  Température maximale sur 12 heures (°C)              float64
 68  Température maximale sur 24 heures (°C)              float64
 69  Température minimale du sol sur 12 heures (en °C)    float64
 70  Latitude                                             float64
 71  Longitude                                            float64
 72  Altitude                                             int64
 73  communes (name)                                      object
 74  communes (code)                                      object
 75  EPCI (name)                                          object
 76  EPCI (code)                                          float64
 77  department (name)                                    object
 78  department (code)                                    object
 79  region (name)                                        object
 80  region (code)                                        float64
 81  mois_de_l_annee                                      int64
dtypes: float64(67), int64(3), object(12)
memory usage: 1.6+ GB
```

In [6]:

```python
# Renommage de colonnes
colonnes_rename = {
    'Date': 'date_heure',
    'Température': 'temperature_K',
    'Humidité': 'humidite_relative',
    'Température (°C)': 'temperature_C',
    'region (name)': 'region'}
df_meteo.rename(columns=colonnes_rename, inplace=True)
df_meteo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2557951 entries, 0 to 2557950
Data columns (total 82 columns):
 #   Column                                                Dtype
---  ------                                                -----
 0   ID OMM station                                        int64
 1   date_heure                                            object
 2   Pression au niveau mer                                float64
 3   Variation de pression en 3 heures                     float64
 4   Type de tendance barométrique                         float64
 5   Direction du vent moyen 10 mn                         float64
 6   Vitesse du vent moyen 10 mn                           float64
 7   temperature_K                                         float64
 8   Point de rosée                                        float64
 9   humidite_relative                                     float64
 10  Visibilité horizontale                                float64
 11  Temps présent                                         float64
 12  Temps passé 1                                         float64
 13  Temps passé 2                                         float64
 14  Nebulosité totale                                     float64
 15  Nébulosité  des nuages de l' étage inférieur          float64
 16  Hauteur de la base des nuages de l'étage inférieur    float64
 17  Type des nuages de l'étage inférieur                  float64
 18  Type des nuages de l'étage moyen                      float64
 19  Type des nuages de l'étage supérieur                  float64
 20  Pression station                                      float64
 21  Niveau barométrique                                   float64
 22  Géopotentiel                                          float64
 23  Variation de pression en 24 heures                    float64
 24  Température minimale sur 12 heures                    float64
 25  Température minimale sur 24 heures                    float64
 26  Température maximale sur 12 heures                    float64
 27  Température maximale sur 24 heures                    float64
 28  Température minimale du sol sur 12 heures             float64
 29  Méthode de mesure Température du thermomètre mouillé  float64
 30  Température du thermomètre mouillé                    float64
 31  Rafale sur les 10 dernières minutes                   float64
 32  Rafales sur une période                               float64
 33  Periode de mesure de la rafale                        float64
 34  Etat du sol                                           float64
 35  Hauteur totale de la couche de neige, glace, autre au sol  float64
 36  Hauteur de la neige fraîche                           float64
 37  Periode de mesure de la neige fraiche                 float64
 38  Précipitations dans la dernière heure                 float64
 39  Précipitations dans les 3 dernières heures            float64
 40  Précipitations dans les 6 dernières heures            float64
 41  Précipitations dans les 12 dernières heures           float64
 42  Précipitations dans les 24 dernières heures           float64
 43  Phénomène spécial 1                                   float64
 44  Phénomène spécial 2                                   float64
 45  Phénomène spécial 3                                   float64
```

```
 46  Phénomène spécial 4                                  float64
 47  Nébulosité couche nuageuse 1                         float64
 48  Type nuage 1                                         float64
 49  Hauteur de base 1                                    float64
 50  Nébulosité couche nuageuse 2                         float64
 51  Type nuage 2                                         float64
 52  Hauteur de base 2                                    float64
 53  Nébulosité couche nuageuse 3                         float64
 54  Type nuage 3                                         float64
 55  Hauteur de base 3                                    float64
 56  Nébulosité couche nuageuse 4                         float64
 57  Type nuage 4                                         float64
 58  Hauteur de base 4                                    float64
 59  Coordonnees                                          object
 60  Nom                                                  object
 61  Type de tendance barométrique.1                      object
 62  Temps passé 1.1                                      object
 63  Temps présent.1                                      object
 64  temperature_C                                        float64
 65  Température minimale sur 12 heures (°C)              float64
 66  Température minimale sur 24 heures (°C)              float64
 67  Température maximale sur 12 heures (°C)              float64
 68  Température maximale sur 24 heures (°C)              float64
 69  Température minimale du sol sur 12 heures (en °C)    float64
 70  Latitude                                             float64
 71  Longitude                                            float64
 72  Altitude                                             int64
 73  communes (name)                                      object
 74  communes (code)                                      object
 75  EPCI (name)                                          object
 76  EPCI (code)                                          float64
 77  department (name)                                    object
 78  department (code)                                    object
 79  region                                               object
 80  region (code)                                        float64
 81  mois_de_l_annee                                      int64
dtypes: float64(67), int64(3), object(12)
memory usage: 1.6+ GB
```

In [7]:

```python
# Filtrage  et affichage de toutes les lignes avec des doublons en fonction des colonnes 'date_heure','region' et 'ID OMM station'
doublons_lignes = df_meteo[df_meteo.duplicated(subset=['date_heure', 'region','ID OMM station'], keep=False)]

# Affichage des doublons avec toutes leurs colonnes
print(doublons_lignes)
```

```
         ID OMM station                date_heure  Pression au niveau mer  \
88109              7471  2016-12-21T16:00:00+01:00                     NaN
152318             7280  2016-05-06T14:00:00+02:00                101290.0
237752             7510  2015-03-29T08:00:00+02:00                102480.0
412763             7280  2016-05-06T14:00:00+02:00                101290.0
555913             7471  2016-12-20T13:00:00+01:00                     NaN
785539             7471  2016-12-21T16:00:00+01:00                     NaN
874912             7510  2015-03-29T08:00:00+02:00                102480.0
896432             7650  2011-01-12T13:00:00+01:00                102170.0
949487             7471  2016-12-20T13:00:00+01:00                     NaN
1116732            7650  2011-01-12T13:00:00+01:00                102170.0

         Variation de pression en 3 heures  Type de tendance barométrique  \
88109                                -20.0                            5.0
152318                              -160.0                            8.0
237752                               -10.0                            8.0
412763                              -160.0                            8.0
555913                               -20.0                            8.0
785539                               -20.0                            5.0
874912                               -10.0                            8.0
896432                                40.0                            0.0
949487                               -20.0                            8.0
1116732                               40.0                            0.0

         Direction du vent moyen 10 mn  Vitesse du vent moyen 10 mn  \
88109                              0.0                          0.0
152318                           170.0                          8.2
237752                           250.0                          3.6
412763                           170.0                          8.1
555913                             0.0                          0.0
785539                             0.0                          0.0
874912                           250.0                          3.7
896432                           320.0                          3.1
949487                             0.0                          0.0
1116732                          320.0                          3.1

         temperature_K  Point de rosée  humidite_relative  ...  Altitude  \
88109           273.35          272.75               96.0  ...       833
152318          294.75          279.45               37.0  ...       219
237752          284.25          282.35               88.0  ...        47
412763          294.75          279.45               37.0  ...       219
555913          274.75          273.85               94.0  ...       833
785539          273.35          272.75               96.0  ...       833
874912          284.25          282.35               88.0  ...        47
896432          285.05          276.25               55.0  ...         9
949487          274.75          273.85               94.0  ...       833
1116732         285.05          276.25               55.0  ...         9

         communes (name)  communes (code)                     EPCI (name)  \
88109          Chaspuzac            43062                CA du Puy-en-Velay
152318             Ouges            21473                   Dijon Métropole
```

```
237752          Mérignac            33281                  Bordeaux Métropole
412763             Ouges            21473                     Dijon Métropole
555913         Chaspuzac            43062                  CA du Puy-en-Velay
785539         Chaspuzac            43062                  CA du Puy-en-Velay
874912          Mérignac            33281                  Bordeaux Métropole
896432         Marignane            13054  Métropole d'Aix-Marseille-Provence
949487         Chaspuzac            43062                  CA du Puy-en-Velay
1116732        Marignane            13054  Métropole d'Aix-Marseille-Provence

         EPCI (code)  department (name)  department (code)  \
88109    200073419.0        Haute-Loire                 43
152318   242100410.0          Côte-d'Or                 21
237752   243300316.0            Gironde                 33
412763   242100410.0          Côte-d'Or                 21
555913   200073419.0        Haute-Loire                 43
785539   200073419.0        Haute-Loire                 43
874912   243300316.0            Gironde                 33
896432   200054807.0   Bouches-du-Rhône                 13
949487   200073419.0        Haute-Loire                 43
1116732  200054807.0   Bouches-du-Rhône                 13

                           region  region (code)  mois_de_l_annee
88109          Auvergne-Rhône-Alpes           84.0               12
152318      Bourgogne-Franche-Comté           27.0                5
237752           Nouvelle-Aquitaine           75.0                3
412763      Bourgogne-Franche-Comté           27.0                5
555913         Auvergne-Rhône-Alpes           84.0               12
785539         Auvergne-Rhône-Alpes           84.0               12
874912           Nouvelle-Aquitaine           75.0                3
896432   Provence-Alpes-Côte d'Azur           93.0                1
949487         Auvergne-Rhône-Alpes           84.0               12
1116732  Provence-Alpes-Côte d'Azur           93.0                1

[10 rows x 82 columns]
```

In [8]:
```python
# Suppression des doublons en conservant seulement la première occurrence
df_meteo = df_meteo.drop_duplicates(subset=['date_heure', 'region', 'ID OMM station'], keep='first')
```

In [9]:
```python
# Filtrage  et affichage de toutes les lignes avec des doublons en fonction des colonnes 'date_heure','region' et 'ID OMM station'
doublons_lignes2 = df_meteo[df_meteo.duplicated(subset=['date_heure', 'region','ID OMM station'], keep=False)]

# Affichage des doublons avec toutes leurs colonnes
print(doublons_lignes2)
```

```
Empty DataFrame
Columns: [ID OMM station, date_heure, Pression au niveau mer, Variation de pression en 3 heures, Type de tendance barométrique, Direction du vent moyen 10 mn, Vite
sse du vent moyen 10 mn, temperature_K, Point de rosée, humidite_relative, Visibilité horizontale, Temps présent, Temps passé 1, Temps passé 2, Nebulosité totale,
Nébulosité  des nuages de l' étage inférieur, Hauteur de la base des nuages de l'étage inférieur, Type des nuages de l'étage inférieur, Type des nuages de l'étage
moyen, Type des nuages de l'étage supérieur, Pression station, Niveau barométrique, Géopotentiel, Variation de pression en 24 heures, Température minimale sur 12 h
eures, Température minimale sur 24 heures, Température maximale sur 12 heures, Température maximale sur 24 heures, Température minimale du sol sur 12 heures, Métho
de de mesure Température du thermomètre mouillé, Température du thermomètre mouillé, Rafale sur les 10 dernières minutes, Rafales sur une période, Periode de mesur
e de la rafale, Etat du sol, Hauteur totale de la couche de neige, glace, autre au sol, Hauteur de la neige fraîche, Periode de mesure de la neige fraiche, Précipi
tations dans la dernière heure, Précipitations dans les 3 dernières heures, Précipitations dans les 6 dernières heures, Précipitations dans les 12 dernières heure
s, Précipitations dans les 24 dernières heures, Phénomène spécial 1, Phénomène spécial 2, Phénomène spécial 3, Phénomène spécial 4, Nébulosité couche nuageuse 1, T
ype nuage 1, Hauteur de base 1, Nébulosité couche nuageuse 2, Type nuage 2, Hauteur de base 2, Nébulosité couche nuageuse 3, Type nuage 3, Hauteur de base 3, Nébul
osité couche nuageuse 4, Type nuage 4, Hauteur de base 4, Coordonnees, Nom, Type de tendance barométrique.1, Temps passé 1.1, Temps présent.1, temperature_C, Tempé
rature minimale sur 12 heures (°C), Température minimale sur 24 heures (°C), Température maximale sur 12 heures (°C), Température maximale sur 24 heures (°C), Temp
érature minimale du sol sur 12 heures (en °C), Latitude, Longitude, Altitude, communes (name), communes (code), EPCI (name), EPCI (code), department (name), depart
ment (code), region, region (code), mois_de_l_annee]
Index: []

[0 rows x 82 columns]
```

In [10]:
```
# Suppression de colonnes non utiles dans cette étude
colonnes_to_drop_humidite = [
    0, 2, 3, 4, 5, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23,
    24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41,
    42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59,
    60, 61, 62, 63, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78
]
df_meteo = df_meteo.drop(df_meteo.columns[colonnes_to_drop_humidite], axis=1)
df_meteo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2557946 entries, 0 to 2557950
Data columns (total 7 columns):
 #   Column                      Dtype
---  ------                      -----
 0   date_heure                  object
 1   Vitesse du vent moyen 10 mn  float64
 2   humidite_relative           float64
 3   temperature_C               float64
 4   region                      object
 5   region (code)               float64
 6   mois_de_l_annee             int64
dtypes: float64(4), int64(1), object(2)
memory usage: 156.1+ MB
```

In [11]:
```
df_meteo['region'].unique()
```

Out[11]:
```
array(['Occitanie', nan, 'Normandie', 'Bretagne',
       "Provence-Alpes-Côte d'Azur", 'Martinique', 'Hauts-de-France',
       'Île-de-France', 'Guyane', 'Corse', 'Grand Est',
       'Nouvelle-Aquitaine', 'Bourgogne-Franche-Comté',
       'Pays de la Loire', 'Auvergne-Rhône-Alpes', 'La Réunion',
       'Centre-Val de Loire', 'Guadeloupe', 'Mayotte',
       'Terres australes et antarctiques françaises', 'Saint-Barthélemy',
       'Saint-Pierre-et-Miquelon'], dtype=object)
```

In [12]:
```
# Suppression des lignes de régions non concernées par cette étude
regions_a_exclure = [
    np.nan, 'Martinique', 'Guyane', 'Corse', 'La Réunion', 'Guadeloupe',
    'Mayotte', 'Saint-Barthélemy', 'Terres australes et antarctiques françaises',
    'Saint-Pierre-et-Miquelon'
]
df_meteo = df_meteo[~df_meteo['region'].isin(regions_a_exclure)]
df_meteo['region'].unique()
```

Out[12]:
```
array(['Occitanie', 'Normandie', 'Bretagne', "Provence-Alpes-Côte d'Azur",
       'Hauts-de-France', 'Île-de-France', 'Grand Est',
       'Nouvelle-Aquitaine', 'Bourgogne-Franche-Comté',
       'Pays de la Loire', 'Auvergne-Rhône-Alpes', 'Centre-Val de Loire'],
      dtype=object)
```

In [13]:
```
# Renommage des valeurs de 'region'
renommage_regions = {
    'Occitanie': 'OCCITANIE',
    'Normandie': 'NORMANDIE',
    'Bretagne': 'BRETAGNE',
    'Provence-Alpes-Côte d\'Azur': 'PROVENCE ALPES COTE D AZUR',
    'Hauts-de-France': 'HAUTS DE FRANCE',
    'Île-de-France': 'ILE DE FRANCE',
    'Grand Est': 'GRAND EST',
    'Nouvelle-Aquitaine': 'NOUVELLE AQUITAINE',
    'Bourgogne-Franche-Comté': 'BOURGOGNE FRANCHE COMTE',
    'Pays de la Loire': 'PAYS DE LA LOIRE',
    'Auvergne-Rhône-Alpes': 'AUVERGNE RHONE ALPES',
    'Centre-Val de Loire': 'CENTRE VAL DE LOIRE'
}
df_meteo['region'] = df_meteo['region'].replace(renommage_regions)
df_meteo['region'].unique()
```

Out[13]:
```
array(['OCCITANIE', 'NORMANDIE', 'BRETAGNE', 'PROVENCE ALPES COTE D AZUR',
       'HAUTS DE FRANCE', 'ILE DE FRANCE', 'GRAND EST',
       'NOUVELLE AQUITAINE', 'BOURGOGNE FRANCHE COMTE',
       'PAYS DE LA LOIRE', 'AUVERGNE RHONE ALPES', 'CENTRE VAL DE LOIRE'],
      dtype=object)
```

In [14]:
```
# Conversion des dates avec gestion des fuseaux horaires
df_meteo["date_heure"] = pd.to_datetime(df_meteo["date_heure"], errors='coerce', utc=True)
```

```
# Conversion à l'heure française
df_meteo["date_heure"] = df_meteo["date_heure"].dt.tz_convert("Europe/Paris")

# Suppression de la mention de fuseau tout en conservant l'heure correcte
df_meteo["date_heure"] = df_meteo["date_heure"].dt.tz_localize(None)

# Extraction de l'année
df_meteo['annee'] = df_meteo['date_heure'].dt.year

# Ajout d'une colonne 'date' avec le format 'yyyy-mm-dd'
df_meteo['date'] = df_meteo['date_heure'].dt.strftime('%Y-%m-%d')

# Ajout d'une colonne 'heure' à l'heure française
df_meteo['heure'] = df_meteo['date_heure'].dt.strftime('%H:%M')

df_meteo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1741602 entries, 0 to 2557950
Data columns (total 10 columns):
 #   Column                       Dtype         
---  ------                       -----         
 0   date_heure                   datetime64[ns]
 1   Vitesse du vent moyen 10 mn  float64       
 2   humidite_relative            float64       
 3   temperature_C                float64       
 4   region                       object        
 5   region (code)                float64       
 6   mois_de_l_annee              int64         
 7   annee                        int32         
 8   date                         object        
 9   heure                        object        
dtypes: datetime64[ns](1), float64(4), int32(1), int64(1), object(3)
memory usage: 139.5+ MB
```

In [15]:
```
# Suppression des valeurs NaT après conversion
df_meteo = df_meteo.dropna(subset=["date_heure"])
```

In [16]:
```
# Filtrage des années pour limiter le dataset de 2013 à 2022 inclus
df_meteo = df_meteo[(df_meteo["annee"] >= 2013) & (df_meteo["annee"] <= 2022)]

# Vérification
print(df_meteo['annee'].unique())
```

```
[2013 2015 2014 2016 2017 2018 2019 2020 2022 2021]
```

In [17]:
```
# recherche des valeurs manquantes
df_meteo.isna().sum()
```

Out[17]:
```
date_heure                        0
Vitesse du vent moyen 10 mn    3117
humidite_relative             10525
temperature_C                  9768
region                            0
region (code)                     0
mois_de_l_annee                   0
annee                             0
date                              0
heure                             0
dtype: int64
```

In [18]:
```python
# Remplacement des NaN par 0 dans tout le dataframe
df_meteo.fillna(0, inplace=True)
df_meteo.isna().sum()
```

Out[18]:
```
date_heure                    0
Vitesse du vent moyen 10 mn   0
humidite_relative             0
temperature_C                 0
region                        0
region (code)                 0
mois_de_l_annee               0
annee                         0
date                          0
heure                         0
dtype: int64
```

In [19]:
```python
# ***************  Ajout des températures ressenties sous l'effet du vent et de l'humidité *****
```

In [20]:
```python
# Fonction
def calcul_temperature_ressentie(df):
    # Conversion vitesse vent m/s en km/h
    df['vent_kmh'] = df['Vitesse du vent moyen 10 mn'] * 3.6

    # Formule du refroidissement éolien (Wind Chill Index)
    wind_chill = (
        13.12
        + 0.6215 * df['temperature_C']
        - 11.37 * np.power(df['vent_kmh'], 0.16)
        + 0.3965 * df['temperature_C'] * np.power(df['vent_kmh'], 0.16)
    )

    # Formule du Heat Index (simplifiée)
    hi_coeffs = [
        -8.78469475556, 1.61139411, 2.33854883889, -0.14611605,
        -0.012308094, -0.0164248277778, 0.002211732, 0.00072546, -0.000003582
    ]
    T = df['temperature_C']
    RH = df['humidite_relative']
```

```
    heat_index = (
        hi_coeffs[0]
        + hi_coeffs[1] * T
        + hi_coeffs[2] * RH
        + hi_coeffs[3] * T * RH
        + hi_coeffs[4] * T**2
        + hi_coeffs[5] * RH**2
        + hi_coeffs[6] * T**2 * RH
        + hi_coeffs[7] * T * RH**2
        + hi_coeffs[8] * T**2 * RH**2
    )

    # Conditions d'application
    conditions = [
        (df['temperature_C'] <= 10) & (df['vent_kmh'] > 4.8),  # Wind Chill
        (df['temperature_C'] >= 27) & (df['humidite_relative'] > 40),  # Heat Index
    ]
    choices = [wind_chill, heat_index]

    # Valeur ressentie par défaut = température réelle
    df['temperature_ressentie'] = np.select(conditions, choices, default=df['temperature_C'])

    return df.drop(columns=['vent_kmh'])
```

In [21]:
```
# Application de la fonction
df_meteo = calcul_temperature_ressentie(df_meteo)
```

In [22]:
```
# Suppression de la colonne 'humidite_relative'
df_meteo = df_meteo.drop(columns =['humidite_relative'])
df_meteo.head()
```

Out[22]:

	date_heure	Vitesse du vent moyen 10 mn	temperature_C	region	region (code)	mois_de_l_annee	annee	date	heure	temperature_ressentie
1788	2013-03-05 04:00:00	14.9	11.8	OCCITANIE	76.0	3	2013	2013-03-05	04:00	11.800000
1790	2013-03-05 07:00:00	1.5	2.8	NORMANDIE	28.0	3	2013	2013-03-05	07:00	1.422594
1791	2013-03-05 07:00:00	0.0	-3.0	GRAND EST	44.0	3	2013	2013-03-05	07:00	-3.000000
1792	2013-03-05 07:00:00	10.3	10.0	AUVERGNE RHONE ALPES	84.0	3	2013	2013-03-05	07:00	6.134590
1793	2013-03-05 07:00:00	13.9	11.1	OCCITANIE	76.0	3	2013	2013-03-05	07:00	11.100000

In [23]:
```
df_meteo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1159540 entries, 1788 to 2557459
Data columns (total 10 columns):
 #   Column                       Non-Null Count    Dtype
---  ------                       --------------    -----
 0   date_heure                   1159540 non-null  datetime64[ns]
 1   Vitesse du vent moyen 10 mn  1159540 non-null  float64
 2   temperature_C                1159540 non-null  float64
 3   region                       1159540 non-null  object
 4   region (code)                1159540 non-null  float64
 5   mois_de_l_annee              1159540 non-null  int64
 6   annee                        1159540 non-null  int32
 7   date                         1159540 non-null  object
 8   heure                        1159540 non-null  object
 9   temperature_ressentie        1159540 non-null  float64
dtypes: datetime64[ns](1), float64(4), int32(1), int64(1), object(3)
memory usage: 92.9+ MB
```

In [24]:
```
# Export
df_meteo.to_csv('df_temperature_ressentie.csv', index=False)
df_meteo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1159540 entries, 1788 to 2557459
Data columns (total 10 columns):
 #   Column                       Non-Null Count    Dtype
---  ------                       --------------    -----
 0   date_heure                   1159540 non-null  datetime64[ns]
 1   Vitesse du vent moyen 10 mn  1159540 non-null  float64
 2   temperature_C                1159540 non-null  float64
 3   region                       1159540 non-null  object
 4   region (code)                1159540 non-null  float64
 5   mois_de_l_annee              1159540 non-null  int64
 6   annee                        1159540 non-null  int32
 7   date                         1159540 non-null  object
 8   heure                        1159540 non-null  object
 9   temperature_ressentie        1159540 non-null  float64
dtypes: datetime64[ns](1), float64(4), int32(1), int64(1), object(3)
memory usage: 92.9+ MB
```

In [25]:
```
# ********************  INTERPOLATION DES DONNEES METEO POUR OBTENIR DES DONNEES PAR PAS DE 30 MINUTES  ************************
# ******************** CALCUL DES MOYENNES ET INTERPOLATION DES DONNÉES ************************
```

In [26]:
```
# Colonnes à interpoler
columns_to_interpolate = ['Vitesse du vent moyen 10 mn', 'temperature_C', 'temperature_ressentie']

# Calcul des moyennes pour chaque combinaison de date_heure et région
df_meteo_grouped = df_meteo.groupby(["date_heure", "region"], as_index=False).mean(numeric_only=True)
```

```
# Création de la plage de dates avec un pas de 30 minutes
date_range = pd.date_range(
    start="2013-01-01 00:00:00",
    end="2022-12-31 23:00:00",
    freq="30min")

# Création du DataFrame de référence
regions = df_meteo['region'].unique()
all_dates = pd.DataFrame({
    "date_heure": np.tile(date_range, len(regions)),
    "region": np.repeat(regions, len(date_range))
})

# Jointure pour aligner les dates avec df_meteo_grouped
df_meteo_aligned = pd.merge(
    all_dates, df_meteo_grouped,
    on=["date_heure", "region"],
    how="left"
)

# Interpolation par région avec affichage d'une barre de progression
tqdm.pandas(desc="Interpolation des données")

# Interpolation des colonnes sélectionnées
interpolated_df = df_meteo_aligned.groupby("region").progress_apply(
    lambda group: group.set_index("date_heure")[columns_to_interpolate]
    .interpolate(method="time", limit_direction="both")
    .reset_index()
)

# Fusion avec les autres colonnes
interpolated_df = pd.merge(
    df_meteo_aligned.drop(columns=columns_to_interpolate),
    interpolated_df,
    on=["date_heure", "region"],
    how="left"
)

# Suppression de colonnes non utiles
colonnes_to_drop_interpolated =['region (code)','mois_de_l_annee','annee']
interpolated_df=interpolated_df.drop(colonnes_to_drop_interpolated, axis = 1)


# Tri et sauvegarde des résultats
df_interpolated = interpolated_df.sort_values(['region', 'date_heure'])
df_interpolated.to_csv("df_meteo_interpolated.csv", index=False)
```

```
Interpolation des données: 100%|██████████| 12/12 [00:00<00:00, 13.96it/s]
```

In [27]:
```
df_interpolated.head()
```

Out[27]:

	date_heure	region	Vitesse du vent moyen 10 mn	temperature_C	temperature_ressentie
525885	2013-01-01 00:00:00	AUVERGNE RHONE ALPES	8.475000	7.375000	3.347231
525886	2013-01-01 00:30:00	AUVERGNE RHONE ALPES	8.475000	7.375000	3.347231
525887	2013-01-01 01:00:00	AUVERGNE RHONE ALPES	8.475000	7.375000	3.347231
525888	2013-01-01 01:30:00	AUVERGNE RHONE ALPES	8.154167	7.416667	3.548545
525889	2013-01-01 02:00:00	AUVERGNE RHONE ALPES	7.833333	7.458333	3.749860

In [28]:
```
df_interpolated.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2103540 entries, 525885 to 1227064
Data columns (total 5 columns):
 #   Column                       Dtype         
---  ------                       -----         
 0   date_heure                   datetime64[ns]
 1   region                       object        
 2   Vitesse du vent moyen 10 mn  float64       
 3   temperature_C                float64       
 4   temperature_ressentie        float64       
dtypes: datetime64[ns](1), float64(3), object(1)
memory usage: 96.3+ MB
```

In [31]:
```
print("Valeurs manquantes:",df_interpolated.isna().sum())
```

```
Valeurs manquantes: date_heure                   0
region                         0
Vitesse du vent moyen 10 mn    0
temperature_C                  0
temperature_ressentie          0
dtype: int64
```