Découverte des données

In [12]:
```
# importer les packages
import os
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [14]:
```
files=[file for file in os.listdir('C:\Users\HP\Downloads\sales_assesment')]
for file in files:
    print(file)
```

```
.ipynb_checkpoints
Sales_April_2019.csv
Sales_August_2019.csv
Sales_December_2019.csv
Sales_February_2019.csv
Sales_January_2019.csv
Sales_July_2019.csv
Sales_June_2019.csv
Sales_March_2019.csv
Sales_May_2019.csv
Sales_November_2019.csv
Sales_October_2019.csv
Sales_September_2019.csv
Untitled.ipynb
```

In [15]:
```
# regrouper ces fichiers dans un seul fichier
path='C:\Users\HP\Downloads\sales_assesment'
all_data=pd.DataFrame()
# j'ai créer une dataframe et la remplir
for file in files:
    current_data=pd.read_csv(path +'/' + file)
    all_data=pd.concat([all_data,current_data])
print(all_data)
```

```
---------------------------------------------------------------------------
PermissionError                           Traceback (most recent call last)
Cell In[15], line 6
      4 # j'ai créer une dataframe et la remplir
      5 for file in files:
----> 6     current_data=pd.read_csv(path +'/' + file)
      7     all_data=pd.concat([all_data,current_data])
      8 print(all_data)

File c:\anaconda3\lib\site-packages\pandas\io\parsers\readers.py:948, in read_csv(filepath_or_buffer, sep, delimiter, header, names, index_col, usecols, dtype, engine, converters, true_values, false_values, skipinitialspace, skiprows, skipfooter, nrows, na_values, keep_default_na, na_filter, verbose, skip_blank_lines, parse_dates, infer_datetime_format, keep_date_col, date_parser, dayfirst, cache_dates, iterator, chunksize, compression, thousands, decimal, lineterminator, quotechar, quoting, doublequote, escapechar, comment, encoding, encoding_errors, dialect, on_bad_lines, delim_whitespace, low_memory, memory_map, float_precision, storage_options, dtype_backend)
    935 kwds_defaults = _refine_defaults_read(
    936     dialect,
    937     delimiter,
    938     error_bad_lines,
    939     warn_bad_lines,
    940     on_bad_lines,
    941     delim_whitespace,
    942     sep,
    943     default_sep,
    944     dtype_backend=dtype_backend,
    945     )
    946 kwds.update(kwds_defaults)
--> 948 return _read(filepath_or_buffer, kwds)

File c:\anaconda3\lib\site-packages\pandas\io\parsers\readers.py:611, in _read(filepath_or_buffer, kwds)
    608 _validate_names(kwds.get("names", None))
    609 # Create the parser.
--> 611 parser = TextFileReader(filepath_or_buffer, **kwds)
    612 if chunksize or iterator:
    613     return parser

File c:\anaconda3\lib\site-packages\pandas\io\parsers\readers.py:1448, in TextFileReader.__init__(self, f, engine, **kwds)
   1445     self.options["has_index_names"] = kwds["has_index_names"]
   1447 self.handles: IOHandles | None = None
-> 1448 self._engine = self._make_engine(f, self.engine)

File c:\anaconda3\lib\site-packages\pandas\io\parsers\readers.py:1705, in TextFileReader._make_engine(self, f, engine)
   1703     if "b" not in mode:
   1704         mode += "b"
-> 1705 self.handles = get_handle(
   1706     f,
   1707     mode,
   1708     encoding=self.options.get("encoding", None),
   1709     compression=self.options.get("compression", None),
   1710     memory_map=self.options.get("memory_map", False),
   1711     is_text=is_text,
   1712     errors=self.options.get("encoding_errors", "strict"),
   1713     storage_options=self.options.get("storage_options", None),
   1714 )
   1715 assert self.handles is not None
   1716 f = self.handles.handle

File c:\anaconda3\lib\site-packages\pandas\io\common.py:863, in get_handle(path_or_buf, mode, encoding, compression, memory_map, is_text, errors, storage_options)
    858 elif isinstance(handle, str):
    859     # Check whether the filename is to be opened in binary mode.
    860     # Binary mode does not support 'encoding' and 'newline'.
    861     if ioargs.encoding and "b" not in ioargs.mode:
    862         # Encoding
--> 863         handle = open(
    864             handle,
    865             ioargs.mode,
    866             encoding=ioargs.encoding,
    867             errors=errors,
    868             newline="",
    869         )
    870     else:
    871         # Binary mode
    872         handle = open(handle, ioargs.mode)

PermissionError: [Errno 13] Permission denied: 'C:\\Users\\HP\\Downloads\\sales_assesment/.ipynb_checkpoints'
```

In [16]:
```
#le code qui est juste et qui marche
path = r'C:\Users\HP\Downloads\sales_assesment'
all_data = pd.DataFrame()  # Créez un DataFrame vide pour stocker les données combinées

# Parcourez tous les fichiers dans le répertoire
for file in os.listdir(path):
    if file.endswith('.csv'):  # Assurez-vous que le fichier est un fichier CSV
        current_data = pd.read_csv(os.path.join(path, file))
        all_data = pd.concat([all_data, current_data], ignore_index=True)
print(all_data)
```

```
       Order ID                           Product  Quantity Ordered  Price Each  \
0        176558                ...
```

In [17]:
```
#convertir la data frame all_data en fichier csv
all_data.to_csv(os.path.join(path,'all_data.csv'))
```

In [18]:
```
all_data.dtypes
```

Out[18]:
```
Order ID            object
Product             object
Quantity Ordered    object
Price Each          object
Order Date          object
Purchase Address    object
dtype: object
```

In [19]:
```
all_data.isnull().sum()
```

Out[19]:
```
Order ID            545
Product             545
Quantity Ordered    545
Price Each          545
Order Date          545
Purchase Address    545
dtype: int64
```

In [20]:
```
#supprimer les valeurs manquantes
all_data=all_data.dropna(how='all')
all_data.shape
```

Out[20]:
```
(186305, 6)
```

what is the month where we achieved the most turnover?

In [21]:
```
all_data
```

Out[22]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215
3	176560	Google Phone	1	600	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001
...	...	...	...	...	...	...
186845	259353	AAA Batteries (4-pack)	3	2.99	09/17/19 20:56	840 Highland St, Los Angeles, CA 90001
186846	259354	iPhone	1	700	09/01/19 16:00	216 Dogwood St, San Francisco, CA 94016
186847	259355	iPhone	1	700	09/23/19 07:39	220 12th St, San Francisco, CA 94016
186848	259356	34in Ultrawide Monitor	1	379.99	09/19/19 17:30	511 Forest St, San Francisco, CA 94016
186849	259357	USB-C Charging Cable	1	11.95	09/30/19 00:18	250 Meadow St, San Francisco, CA 94016

186305 rows × 6 columns

In [22]:
```
def month(x):
    return x.split('/')[0]
month('12/30/19 00:01')
```

Out[23]:
```
'12'
```

In [23]:
```
all_data['Month']=all_data['Order Date'].apply(month)
all_data
```

Out[23]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001	04
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	04
3	176560	Google Phone	1	600	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	04
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	04
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001	04
...	...	...	...	...	...	...	...
186845	259353	AAA Batteries (4-pack)	3	2.99	09/17/19 20:56	840 Highland St, Los Angeles, CA 90001	09
186846	259354	iPhone	1	700	09/01/19 16:00	216 Dogwood St, San Francisco, CA 94016	09
186847	259355	iPhone	1	700	09/23/19 07:39	220 12th St, San Francisco, CA 94016	09
186848	259356	34in Ultrawide Monitor	1	379.99	09/19/19 17:30	511 Forest St, San Francisco, CA 94016	09
186849	259357	USB-C Charging Cable	1	11.95	09/30/19 00:18	250 Meadow St, San Francisco, CA 94016	09

186305 rows × 7 columns

In [24]:
```
all_data['Month'].unique()
```

Out[24]:
```
array(['04', '05', 'Order Date', '08', '09', '12', '01', '02', '03', '07',
       '06', '11', '10'], dtype=object)
```

In [25]:
```
all_data=all_data[all_data['Month']!='Order Date']
all_data['Month'].unique()
```

Out[25]:
```
array(['04', '05', '08', '09', '12', '01', '02', '03', '07', '06', '11',
       '10'], dtype=object)
```

In [26]:
```
all_data.dtypes
```

Out[26]:
```
Order ID            object
Product             object
Quantity Ordered    object
Price Each          object
Order Date          object
Purchase Address    object
Month               object
dtype: object
```

In [27]:
```
all_data['Month']=all_data['Month'].astype(int)
all_data.dtypes
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_11610\1533290404.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  all_data['Month']=all_data['Month'].astype(int)
```

Out[27]:
```
Order ID            object
Product             object
Quantity Ordered    object
Price Each          object
Order Date          object
Purchase Address    object
Month               int32
dtype: object
```

In [28]:
```
all_data['Price Each']=all_data['Price Each'].astype(float)
all_data['Quantity Ordered']=all_data['Quantity Ordered'].astype(int)
all_data.dtypes
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_11610\2148845491.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  all_data['Price Each']=all_data['Price Each'].astype(float)
C:\Users\HP\AppData\Local\Temp\ipykernel_11610\2148845491.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  all_data['Quantity Ordered']=all_data['Quantity Ordered'].astype(int)
```

Out[28]:
```
Order ID            object
Product             object
Quantity Ordered    int32
Price Each        float64
Order Date          object
Purchase Address    object
Month               int32
dtype: object
```

In [29]:
```
all_data['Sales']=all_data['Quantity Ordered']*all_data['Price Each']
all_data
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_11610\3208771603.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  all_data['Sales']=all_data['Quantity Ordered']*all_data['Price Each']
```

Out[30]:

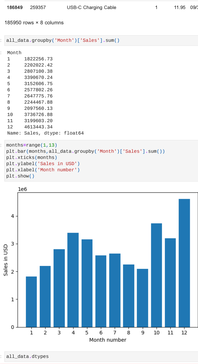
	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Sales
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001	4	23.90
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	4	99.99
3	176560	Google Phone	1	600.00	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	600.00
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	11.99
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001	4	11.99
...	...	...	...	...	...	...	...	...
186845	259353	AAA Batteries (4-pack)	3	2.99	09/17/19 20:56	840 Highland St, Los Angeles, CA 90001	9	8.97
186846	259354	iPhone	1	700.00	09/01/19 16:00	216 Dogwood St, San Francisco, CA 94016	9	700.00
186847	259355	iPhone	1	700.00	09/23/19 07:39	220 12th St, San Francisco, CA 94016	9	700.00
186848	259356	34in Ultrawide Monitor	1	379.99	09/19/19 17:30	511 Forest St, San Francisco, CA 94016	9	379.99
186849	259357	USB-C Charging Cable	1	11.95	09/30/19 00:18	250 Meadow St, San Francisco, CA 94016	9	11.95

185950 rows × 8 columns

In [31]:
```
all_data.groupby('Month')['Sales'].sum()
```

Out[31]:
```
Month
1     1822256.73
2     2202022.42
3     2807100.38
4     3390670.24
5     3152606.75
6     2577802.26
7     2647775.76
8     2244467.88
9     2097560.13
10    3736726.88
11    3190603.20
12    4613443.34
Name: Sales, dtype: float64
```

In [34]:
```
month=range(1,13)
plt.bar(months,all_data.groupby('Month')['Sales'].sum())
plt.xticks(months)
plt.ylabel('Sales in USD')
plt.xlabel('Month number')
plt.show()
```



In [4]:
```
all_data.dtypes
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[4], line 1
----> 1 all_data.dtypes

NameError: name 'all_data' is not defined
```

In [ ]: