

```
pip install pandas
```

Defaulting to user installation because normal site-packages is not writeable

Requirement already satisfied: pandas in c:\users\user\appdata\roaming\python\python313\site-packages (2.3.3)

Requirement already satisfied: numpy>=1.26.0 in c:\users\user\appdata\roaming\python\python313\site-packages (from pandas) (2.3.3)

Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\user\appdata\roaming\python\python313\site-packages (from pandas) (2.9.0.post0)

Requirement already satisfied: pytz>=2020.1 in c:\users\user\appdata\roaming\python\python313\site-packages (from pandas) (2025.2)

Requirement already satisfied: tzdata>=2022.7 in c:\users\user\appdata\roaming\python\python313\site-packages (from pandas) (2025.2)

Requirement already satisfied: six>=1.5 in c:\users\user\appdata\roaming\python\python313\site-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)

Note: you may need to restart the kernel to use updated packages.

```
import pandas as pd
```

```
df=pd.read_csv(r"c:\Users\user\Downloads\sales_performance_dataset(1).csv")
```

```
df
```

	Employee_ID	Employee_Name	Department	Experience_Years
Monthly_Sales \				
0	1001	Ravi	Finance	15
41834				
1	1002	Sneha	Marketing	13
38047				
2	1003	Amit	IT	1
46105				
3	1004	Priya	Marketing	9
95766				
4	1005	Karan	Marketing	7
35707				
..
...				
95	1096	Megha	Finance	12
93656				
96	1097	Pritam	Finance	12
59384				
97	1098	Ramesh	Marketing	4
67254				
98	1099	Shivani	Sales	14
41918				
99	1100	Niraj	Marketing	14

105981

```
Customer_Satisfaction
0      10
1       3
2       7
3      10
4       9
..     ...
95      6
96      7
97      2
98     10
99      2
```

[100 rows x 6 columns]

```
print(df.head())
```

```
Employee_ID Employee_Name Department Experience_Years
Monthly_Sales \
0      1001          Ravi    Finance             15
41834
1      1002          Sneha  Marketing             13
38047
2      1003          Amit      IT              1
46105
3      1004          Priya  Marketing             9
95766
4      1005          Karan  Marketing             7
35707
```

```
Customer_Satisfaction
0      10
1       3
2       7
3      10
4       9
```

```
print(df.tail())
```

```
Employee_ID Employee_Name Department Experience_Years
Monthly_Sales \
95      1096          Megha    Finance             12
93656
96      1097          Pritam    Finance             12
59384
97      1098          Ramesh  Marketing             4
67254
98      1099          Shivani    Sales             14
```

```
41918
99      1100      Niraj  Marketing      14
105981
```

```
      Customer_Satisfaction
95      6
96      7
97      2
98     10
99      2
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 100 entries, 0 to 99
```

```
Data columns (total 6 columns):
```

#	Column	Non-Null Count	Dtype
0	Employee_ID	100 non-null	int64
1	Employee_Name	100 non-null	object
2	Department	100 non-null	object
3	Experience_Years	100 non-null	int64
4	Monthly_Sales	100 non-null	int64
5	Customer_Satisfaction	100 non-null	int64

```
dtypes: int64(4), object(2)
```

```
memory usage: 4.8+ KB
```

```
df.describe()
```

	Employee_ID	Experience_Years	Monthly_Sales
Customer_Satisfaction			
count	100.000000	100.000000	100.000000
mean	1050.500000	8.450000	72030.750000
std	29.011492	4.349329	30111.719996
min	1001.000000	1.000000	20854.000000
25%	1025.750000	5.000000	44832.000000
50%	1050.500000	8.500000	72697.500000
75%	1075.250000	12.000000	101270.000000
max	1100.000000	15.000000	118506.000000

```
df.shape
```

```
(100, 6)
```

```
df.columns
```

```
Index(['Employee_ID', 'Employee_Name', 'Department',  
      'Experience_Years',  
      'Monthly_Sales', 'Customer_Satisfaction'],  
      dtype='object')
```

```
df.dtypes
```

```
Employee_ID      int64  
Employee_Name    object  
Department       object  
Experience_Years  int64  
Monthly_Sales    int64  
Customer_Satisfaction  int64  
dtype: object
```

```
df_dropped=df.drop(columns=['Employee_ID'])  
df_dropped
```

	Employee_Name	Department	Experience_Years	Monthly_Sales	\
0	Ravi	Finance	15	41834	
1	Sneha	Marketing	13	38047	
2	Amit	IT	1	46105	
3	Priya	Marketing	9	95766	
4	Karan	Marketing	7	35707	
..	
95	Megha	Finance	12	93656	
96	Pritam	Finance	12	59384	
97	Ramesh	Marketing	4	67254	
98	Shivani	Sales	14	41918	
99	Niraj	Marketing	14	105981	

	Customer_Satisfaction
0	10
1	3
2	7
3	10
4	9
..	...
95	6
96	7
97	2
98	10
99	2

```
[100 rows x 5 columns]
```

```
df_renamed=df.rename(columns={'Employee_Name':'Full name'})  
df_renamed
```

	Employee_ID	Full name	Department	Experience_Years	Monthly_Sales
\					
0	1001	Ravi	Finance	15	41834
1	1002	Sneha	Marketing	13	38047
2	1003	Amit	IT	1	46105
3	1004	Priya	Marketing	9	95766
4	1005	Karan	Marketing	7	35707
..
95	1096	Megha	Finance	12	93656
96	1097	Pritam	Finance	12	59384
97	1098	Ramesh	Marketing	4	67254
98	1099	Shivani	Sales	14	41918
99	1100	Niraj	Marketing	14	105981

	Customer_Satisfaction
0	10
1	3
2	7
3	10
4	9
..	...
95	6
96	7
97	2
98	10
99	2

[100 rows x 6 columns]

```
df_sorted=df.sort_values(by='Experience_Years')
df_sorted
```

	Employee_ID	Employee_Name	Department	Experience_Years
Monthly_Sales \				
2	1003	Amit	IT	1
46105				
7	1008	Arjun	IT	1
43776				
14	1015	Anjali	HR	1
29474				

18	1019	Suman	HR	1
114856				
31	1032	Monika	HR	1
71005				
..
...				
46	1047	Shweta	HR	15
110084				
58	1059	Sameer	Finance	15
118506				
59	1060	Sonia	Finance	15
32688				
65	1066	Rehan	Finance	15
109045				
76	1077	Prakash	HR	15
69811				

Customer_Satisfaction	
2	7
7	2
14	3
18	8
31	9
..	...
46	8
58	3
59	6
65	5
76	8

[100 rows x 6 columns]

```
df_filled=df.fillna(0)
df_filled
```

Employee_ID	Employee_Name	Department	Experience_Years	
Monthly_Sales \				
0	1001	Ravi	Finance	15
41834				
1	1002	Sneha	Marketing	13
38047				
2	1003	Amit	IT	1
46105				
3	1004	Priya	Marketing	9
95766				
4	1005	Karan	Marketing	7
35707				
..
...				
95	1096	Megha	Finance	12

```

93656
96      1097      Pritam      Finance      12
59384
97      1098      Ramesh      Marketing     4
67254
98      1099      Shivani      Sales      14
41918
99      1100      Niraj      Marketing     14
105981

```

```

      Customer_Satisfaction
0              10
1               3
2               7
3              10
4               9
..            ...
95              6
96              7
97              2
98             10
99              2

```

```
[100 rows x 6 columns]
```

```

df_unique=df.drop_duplicates()
df_unique

```

```

      Employee_ID Employee_Name Department  Experience_Years
Monthly_Sales \
0      1001      Ravi      Finance      15
41834
1      1002      Sneha      Marketing     13
38047
2      1003      Amit      IT           1
46105
3      1004      Priya      Marketing     9
95766
4      1005      Karan      Marketing     7
35707
..      ...      ...      ...      ...
...
95      1096      Megha      Finance      12
93656
96      1097      Pritam      Finance      12
59384
97      1098      Ramesh      Marketing     4
67254
98      1099      Shivani      Sales      14
41918

```

```
99          1100          Niraj  Marketing          14
105981
```

```
Customer_Satisfaction
```

```
0          10
1           3
2           7
3          10
4           9
..         ...
95          6
96          7
97          2
98         10
99          2
```

```
[100 rows x 6 columns]
```

```
df_replaced=df.replace({'Ravi':'Raveena'})
df_replaced
```

```
Employee_ID Employee_Name Department Experience_Years
Monthly_Sales \
0          1001          Raveena    Finance          15
41834
1          1002          Sneha    Marketing          13
38047
2          1003          Amit      IT              1
46105
3          1004          Priya    Marketing          9
95766
4          1005          Karan    Marketing          7
35707
..         ...         ...         ...         ...
...
95          1096          Megha    Finance          12
93656
96          1097          Pritam    Finance          12
59384
97          1098          Ramesh    Marketing          4
67254
98          1099          Shivani    Sales          14
41918
99          1100          Niraj    Marketing          14
105981
```

```
Customer_Satisfaction
```

```
0          10
1           3
2           7
```

```

3          10
4           9
..         ...
95          6
96          7
97          2
98         10
99          2

```

```
[100 rows x 6 columns]
```

```

grouped_df=df.groupby('Employee_Name').sum()
grouped_df

```

	Employee_ID	Department	Experience_Years	Monthly_Sales
\ Employee_Name				
Aarav	1069	HR	4	42671
Aditya	1041	Sales	6	87172
Alok	1064	Support	13	72083
Amit	1003	IT	1	46105
Anaya	1090	Marketing	14	91295
...
Varun	1048	Finance	14	58623
Vikas	1031	IT	2	74384
Vinay	1095	Finance	12	28155
Vivek	1012	IT	15	26776
Yash	1035	Support	14	82003

	Customer_Satisfaction
Employee_Name	
Aarav	3
Aditya	4
Alok	1
Amit	7
Anaya	6
...	...
Varun	8
Vikas	3

Vinay	2
Vivek	7
Yash	2

[100 rows x 5 columns]

```
agg_df=df.groupby('Employee_Name').agg({'Experience_Years':'mean'})
agg_df
```

Employee_Name	Experience_Years
Aarav	4.0
Aditya	6.0
Alok	13.0
Amit	1.0
Anaya	14.0
...	...
Varun	14.0
Vikas	2.0
Vinay	12.0
Vivek	15.0
Yash	14.0

[100 rows x 1 columns]

```
count_df=df.groupby('Employee_Name').count()
count_df
```

Monthly_Sales \ Employee_Name	Employee_ID	Department	Experience_Years
Aarav	1	1	1
1			
Aditya	1	1	1
1			
Alok	1	1	1
1			
Amit	1	1	1
1			
Anaya	1	1	1
1			
...
.			
Varun	1	1	1
1			
Vikas	1	1	1
1			
Vinay	1	1	1
1			

Vivek	1	1	1
1			
Yash	1	1	1
1			

Customer_Satisfaction

Employee_Name	
Aarav	1
Aditya	1
Alok	1
Amit	1
Anaya	1
...	...
Varun	1
Vikas	1
Vinay	1
Vivek	1
Yash	1

[100 rows x 5 columns]

```
sum_df=df.groupby('Employee_Name').sum()
sum_df
```

	Employee_ID	Department	Experience_Years	Monthly_Sales
\ Employee_Name				
Aarav	1069	HR	4	42671
Aditya	1041	Sales	6	87172
Alok	1064	Support	13	72083
Amit	1003	IT	1	46105
Anaya	1090	Marketing	14	91295
...
Varun	1048	Finance	14	58623
Vikas	1031	IT	2	74384
Vinay	1095	Finance	12	28155
Vivek	1012	IT	15	26776
Yash	1035	Support	14	82003

	Customer_Satisfaction
Employee_Name	
Aarav	3
Aditya	4
Alok	1
Amit	7
Anaya	6
...	...
Varun	8
Vikas	3
Vinay	2
Vivek	7
Yash	2

[100 rows x 5 columns]

df.dtypes

Employee_ID	int64
Employee_Name	object
Department	object
Experience_Years	int64
Monthly_Sales	int64
Customer_Satisfaction	int64

dtype: object

```
mean_df=df.groupby("Department")["Experience_Years"].mean()
mean_df
```

Department	
Finance	9.960000
HR	8.052632
IT	7.090909
Marketing	8.411765
Sales	7.909091
Support	7.941176

Name: Experience_Years, dtype: float64

```
max_df=df.groupby('Experience_Years').max()
min_df=df.groupby('Experience_Years').min()
max_df
min_df
```

	Employee_ID	Employee_Name	Department	Monthly_Sales
1	1003	Amit	Finance	29474
2	1031	Tanvi	IT	28680

3	1014	Deepa	IT	22911
4	1069	Aarav	HR	42671
5	1021	Ankit	HR	20854
6	1041	Aditya	HR	57504
7	1005	Asha	Finance	33545
8	1007	Harsh	Finance	43664
9	1004	Bhavana	Finance	41976
10	1022	Diya	Finance	45342
11	1013	Chirag	Finance	22811
12	1009	Megha	Finance	21802
13	1002	Alok	Finance	38047
14	1035	Anaya	Finance	41918
15	1001	Prakash	Finance	26776

Customer_Satisfaction
Experience_Years

1	1
2	2
3	1
4	1
5	1
6	1
7	3
8	1
9	1
10	1
11	1
12	1
13	1
14	2
15	3

df_cleaned=df.dropna()
df_cleaned

Employee_ID	Employee_Name	Department	Experience_Years
Monthly_Sales \			
0	1001	Ravi	Finance
			15

41834				
1	1002	Sneha	Marketing	13
38047				
2	1003	Amit	IT	1
46105				
3	1004	Priya	Marketing	9
95766				
4	1005	Karan	Marketing	7
35707				
..
...				
95	1096	Megha	Finance	12
93656				
96	1097	Pritam	Finance	12
59384				
97	1098	Ramesh	Marketing	4
67254				
98	1099	Shivani	Sales	14
41918				
99	1100	Niraj	Marketing	14
105981				

Customer_Satisfaction	
0	10
1	3
2	7
3	10
4	9
..	...
95	6
96	7
97	2
98	10
99	2

[100 rows x 6 columns]

```
df_filled=df.fillna(0)
df_filled
```

Employee_ID	Employee_Name	Department	Experience_Years	
Monthly_Sales \				
0	1001	Ravi	Finance	15
41834				
1	1002	Sneha	Marketing	13
38047				
2	1003	Amit	IT	1
46105				
3	1004	Priya	Marketing	9
95766				

4	1005	Karan	Marketing	7
35707				
..
...				
95	1096	Megha	Finance	12
93656				
96	1097	Pritam	Finance	12
59384				
97	1098	Ramesh	Marketing	4
67254				
98	1099	Shivani	Sales	14
41918				
99	1100	Niraj	Marketing	14
105981				

Customer_Satisfaction	
0	10
1	3
2	7
3	10
4	9
..	...
95	6
96	7
97	2
98	10
99	2

[100 rows x 6 columns]

```
df_replaced=df.replace({'NaN':0})
df_replaced
```

Employee_ID	Employee_Name	Department	Experience_Years
Monthly_Sales \			
0	1001	Ravi	Finance
41834			15
1	1002	Sneha	Marketing
38047			13
2	1003	Amit	IT
46105			1
3	1004	Priya	Marketing
95766			9
4	1005	Karan	Marketing
35707			7
..
...			
95	1096	Megha	Finance
93656			12
96	1097	Pritam	Finance
			12

```

59384
97      1098      Ramesh  Marketing      4
67254
98      1099      Shivani    Sales      14
41918
99      1100      Niraj    Marketing     14
105981

```

```

      Customer_Satisfaction
0              10
1               3
2               7
3              10
4               9
..            ...
95              6
96              7
97              2
98             10
99              2

```

```
[100 rows x 6 columns]
```

```

df['Contains_a']=df['Employee_Name'].str.contains('A')
df

```

```

      Employee_ID Employee_Name Department  Experience_Years
Monthly_Sales \
0      1001      Ravi      Finance      15
41834
1      1002      Sneha  Marketing      13
38047
2      1003      Amit      IT      1
46105
3      1004      Priya  Marketing      9
95766
4      1005      Karan  Marketing      7
35707
..      ...      ...      ...      ...
...
95      1096      Megha    Finance      12
93656
96      1097      Pritam    Finance      12
59384
97      1098      Ramesh  Marketing      4
67254
98      1099      Shivani    Sales      14
41918
99      1100      Niraj    Marketing     14
105981

```

	Customer_Satisfaction	Contains_a
0	10	False
1	3	False
2	7	True
3	10	False
4	9	False
..
95	6	False
96	7	False
97	2	False
98	10	False
99	2	False

[100 rows x 7 columns]

```
counts=df['Employee_Name'].value_counts()
counts
```

Employee_Name

Ravi	1
Sneha	1
Amit	1
Priya	1
Karan	1

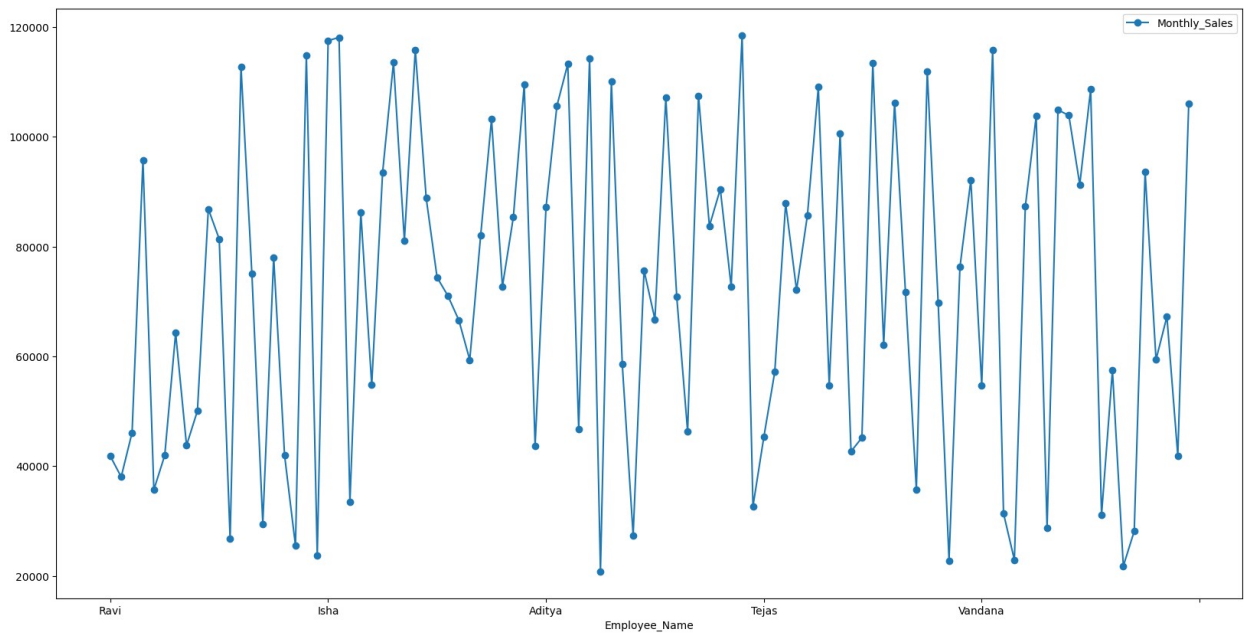
..

Megha	1
Pritam	1
Ramesh	1
Shivani	1
Niraj	1

Name: count, Length: 100, dtype: int64

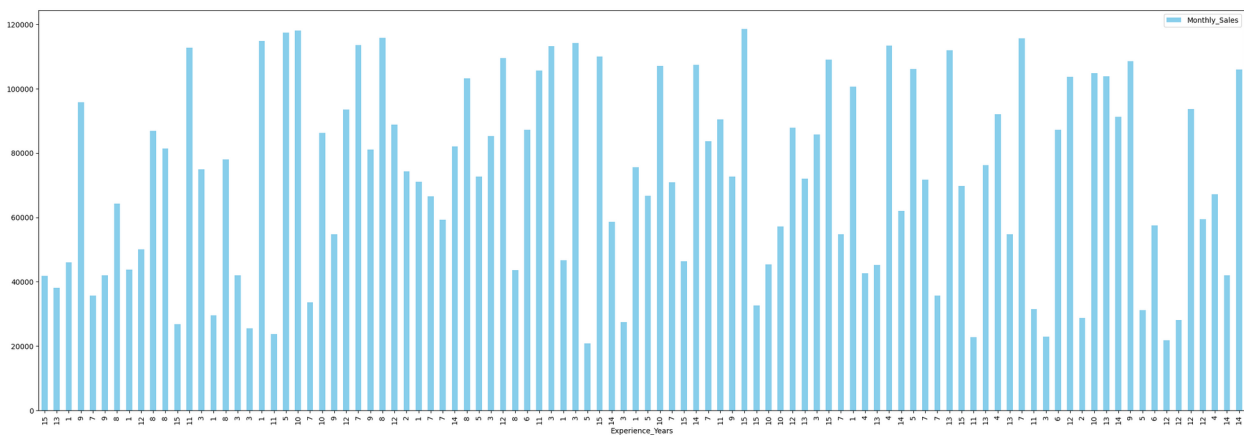
```
import matplotlib.pyplot as plt
df.plot(x='Employee_Name', y='Monthly_Sales', kind='line',
marker='o',figsize=(20,10))
```

<Axes: xlabel='Employee_Name'>



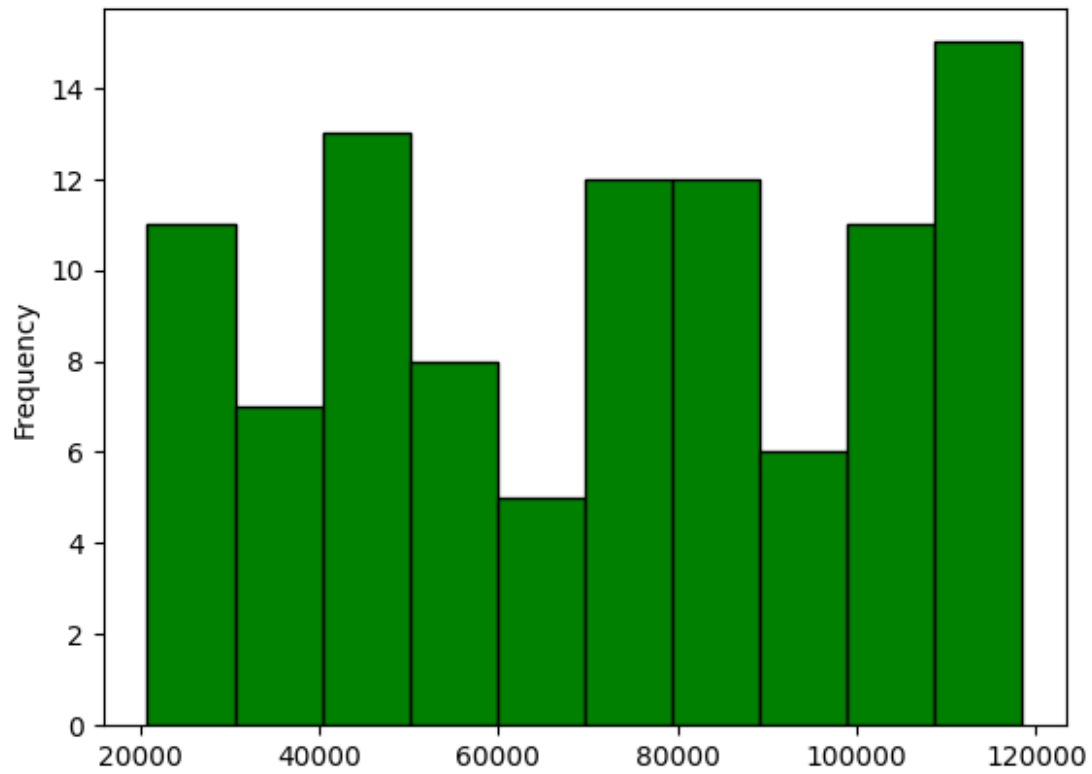
```
df.plot(x='Experience_Years', y='Monthly_Sales', kind='bar',
color='skyblue',figsize=(30,10))
```

```
<Axes: xlabel='Experience_Years'>
```



```
df['Monthly_Sales'].plot(kind='hist',color='green', edgecolor='black')
```

```
<Axes: ylabel='Frequency'>
```



```
df.plot(x='Monthly_Sales', y='Experience_Years', kind='scatter',  
color='red')
```

```
<Axes: xlabel='Monthly_Sales', ylabel='Experience_Years'>
```

