How to Add Axis Labels to Plots in Pandas (With Examples)

Authored by

stats writer

November 24, 2025

Adding axis labels to plots in Pandas is easy with the built-in methods set_xlabel and set_ylabel for the x and y-axis respectively. These methods can be used with a variety of plot types, including line plots, bar plots, and histograms. Examples of how to add labels to plots in Pandas are provided below to demonstrate how to create meaningful visualizations.

You can use the following basic syntax to add axis labels to a plot in pandas:

df.plot(xlabel='X-Axis Label', ylabel='Y-Axis Label')

The following example shows how to use this syntax in practice.

Example: Add Axis Labels to Plot in Pandas

Suppose we have the following pandas DataFrame that shows the total sales made at three stores during consecutive days:

import pandas as pd

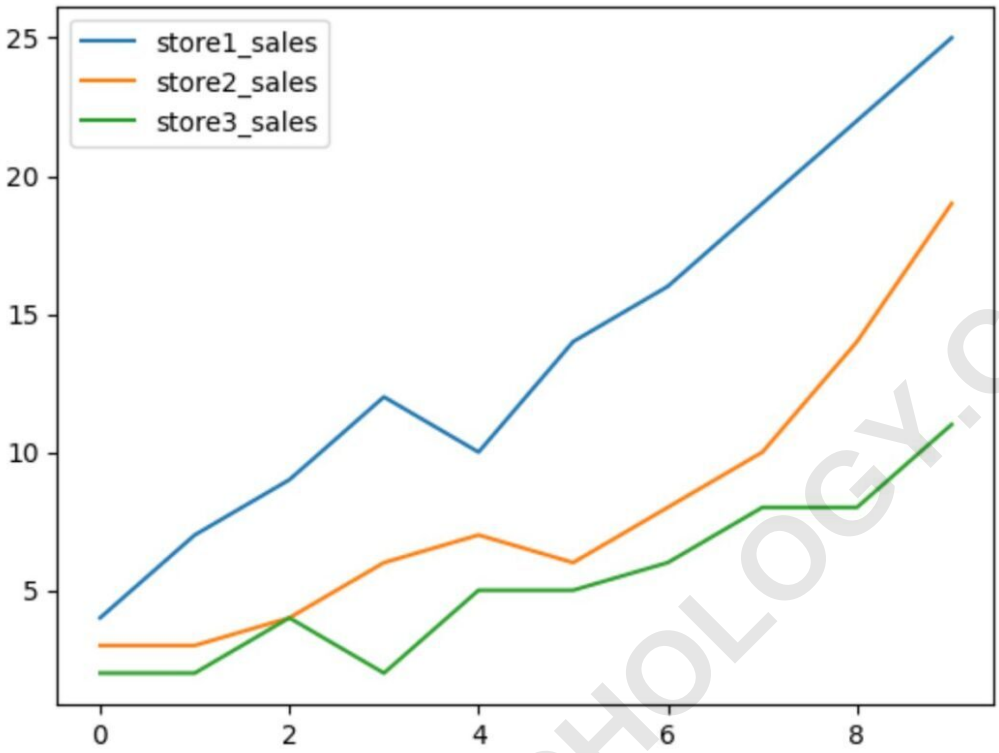
#create DataFrame
df = pd.DataFrame({'store1_sales': ,
'store2_sales': ,
'store3_sales': })

#view DataFrame
print(df)

store1_sales store2_sales store3_sales
0 4 3 2
1 7 3 2
2 9 4 4
3 12 6 2
4 10 7 5
5 14 6 5
6 16 8 6
7 19 10 8
8 22 14 8
9 25 19 11

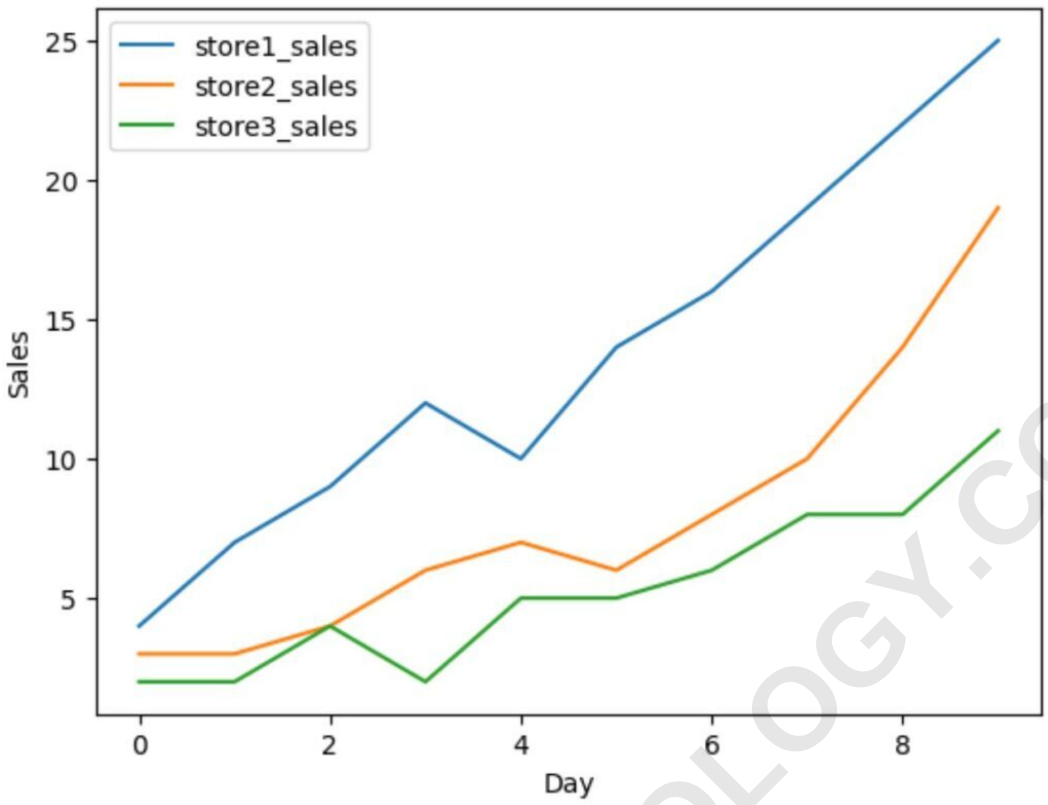
If we create a plot to visualize the sales by store, the pandas **plot()** function will not add axis labels to the plot by default:

#plot sales by store
df.plot()



To add axis labels, we must use the **xlabel** and **ylabel** arguments in the **plot()** function:

#plot sales by store, add axis labels
df.plot(xlabel='Day', ylabel='Sales')



Notice that the x-axis and y-axis now have the labels that we specified within the **plot()** function.

Note that you don't have to use both the **xlabel** and **ylabel** arguments.

For example, you may choose to only add a label to the y-axis:

#plot sales by store, add label to y-axis only
df.plot(ylabel='Sales')