

## ✓ LOADING DATASETS

```
import pandas as pd

# Initialize data
data = {
    'USN': ['1A18CS001', '1A18CS002', '1A18CS003', '1A18CS004', '1A18CS005'],
    'Name': ['Alice', 'Bob', 'Charlie', 'David', 'Eve'],
    'Marks': [85, 78, 92, 88, 75]
}

# Create DataFrame
df = pd.DataFrame(data)

print(df)
```

```
↗
```

	USN	Name	Marks
0	1A18CS001	Alice	85
1	1A18CS002	Bob	78
2	1A18CS003	Charlie	92
3	1A18CS004	David	88
4	1A18CS005	Eve	75

```
from sklearn.datasets import load_diabetes
```

```
diabetes_data = load_diabetes()
```

```
import pandas as pd
df = pd.DataFrame(diabetes_data.data, columns=diabetes_data.feature_names)
df['target'] = diabetes_data.target

print(df.head())
```

```
↗
```

	age	sex	bmi	bp	s1	s2	s3	\
0	0.038076	0.050680	0.061696	0.021872	-0.044223	-0.034821	-0.043401	
1	-0.001882	-0.044642	-0.051474	-0.026328	-0.008449	-0.019163	0.074412	
2	0.085299	0.050680	0.044451	-0.005670	-0.045599	-0.034194	-0.032356	
3	-0.089063	-0.044642	-0.011595	-0.036656	0.012191	0.024991	-0.036038	
4	0.005383	-0.044642	-0.036385	0.021872	0.003935	0.015596	0.008142	

	s4	s5	s6	target
0	-0.002592	0.019907	-0.017646	151.0
1	-0.039493	-0.068332	-0.092204	75.0
2	-0.002592	0.002861	-0.025930	141.0
3	0.034309	0.022688	-0.009362	206.0
4	-0.002592	-0.031988	-0.046641	135.0

```
import pandas as pd
```

```
df = pd.read_csv('sample_sales_data.csv')
```

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

```
↗
```

	Product	Quantity	Price	Sales	Region
0	Laptop	5	1000	5000	North
1	Mouse	15	20	300	West
2	Keyboard	10	50	500	East
3	Monitor	8	200	1600	South
4	Laptop	12	950	11400	North

```
import pandas as pd
```

```
df = pd.read_csv('Dataset of Diabetes .csv')
```

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

```
↗
```

	ID	No_Patien	Gender	AGE	Urea	Cr	HbA1c	Chol	TG	HDL	LDL	VLDL	\
0	502	17975	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	
1	735	34221	M	26	4.5	62	4.9	3.7	1.4	1.1	2.1	0.6	
2	420	47975	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5	

3	680	87656	F	50	4.7	46	4.9	4.2	0.9	2.4	1.4	0.5
4	504	34223	M	33	7.1	46	4.9	4.9	1.0	0.8	2.0	0.4

BMI CLASS

0	24.0	N
1	23.0	N
2	24.0	N
3	24.0	N
4	21.0	N

✓ STOCK MARKET ANALYSIS

```
import yfinance as yf
import pandas as pd
import matplotlib.pyplot as plt

tickers = ["HDFCBANK.NS", "ICICIBANK.NS", "KOTAKBANK.NS"]

# Fetch historical data for the last 1 year
data = yf.download(tickers, start="2024-01-01", end="2024-12-30", group_by='ticker')

# Display the first 5 rows of the dataset
print("First 5 rows of the dataset:")
print(data.head())
```

3 of 3 completed

First 5 rows of the dataset:

Ticker	KOTAKBANK.NS					
Price	Open	High	Low	Close	Volume	
Date						
2024-01-01	1906.909954	1916.899006	1891.027338	1907.059814	1425902	
2024-01-02	1905.911108	1905.911108	1858.063525	1863.008179	5120796	
2024-01-03	1861.959234	1867.952665	1845.627158	1863.857178	3781515	
2024-01-04	1869.451068	1869.451068	1858.513105	1861.559692	2865766	
2024-01-05	1863.457575	1867.852782	1839.383985	1845.577148	7799341	

Ticker	HDFCBANK.NS					
Price	Open	High	Low	Close	Volume	
Date						
2024-01-01	1683.017598	1686.125187	1669.206199	1675.223999	7119843	
2024-01-02	1675.914685	1679.860799	1665.950651	1676.210571	14621046	
2024-01-03	1679.071480	1681.735059	1646.466666	1650.363525	14194881	
2024-01-04	1655.394910	1672.116520	1648.193203	1668.071777	13367028	
2024-01-05	1664.421596	1681.932477	1645.628180	1659.538208	15944735	

Ticker	ICICIBANK.NS					
Price	Open	High	Low	Close	Volume	
Date						
2024-01-01	983.086778	996.273246	982.541485	990.869812	7683792	
2024-01-02	988.490253	989.134730	971.883221	973.866150	16263825	
2024-01-03	976.295294	979.567116	966.777197	975.650818	16826752	
2024-01-04	977.980767	980.707295	973.519176	978.724365	22789140	
2024-01-05	979.567084	989.779158	975.402920	985.218445	14875499	

```
hdfc_data = data['HDFCBANK.NS']
hdfc_data['Daily Return'] = hdfc_data['Close'].pct_change()

ic_data = data['ICICIBANK.NS']
ic_data['Daily Return'] = ic_data['Close'].pct_change()

kb_data = data['KOTAKBANK.NS']
kb_data['Daily Return'] = kb_data['Close'].pct_change()
```

<ipython-input-14-8f8743af89e3>: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](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
hdfc\_data['Daily Return'] = hdfc\_data['Close'].pct\_change()

<ipython-input-14-8f8743af89e3>:5: 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

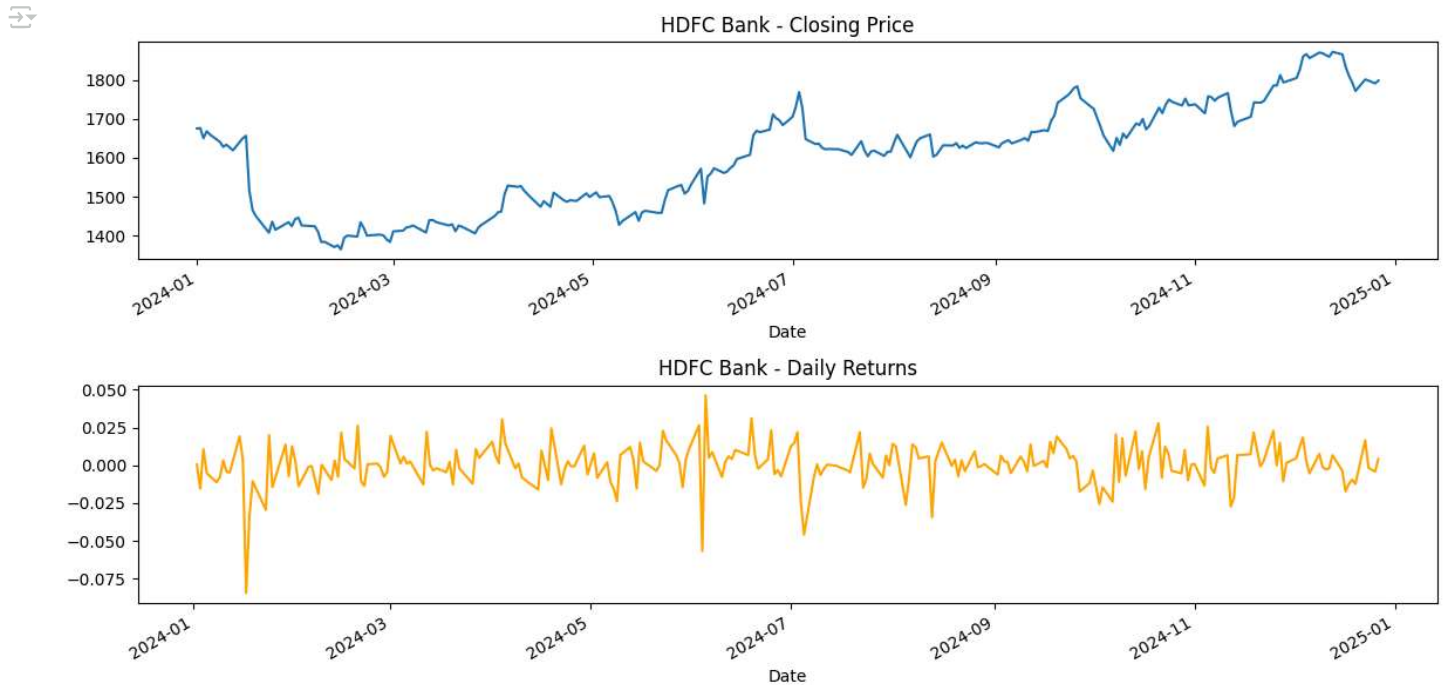
See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
ic\_data['Daily Return'] = ic\_data['Close'].pct\_change()

<ipython-input-14-8f8743af89e3>:8: 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](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
`kb_data['Daily Return'] = kb_data['Close'].pct_change()`

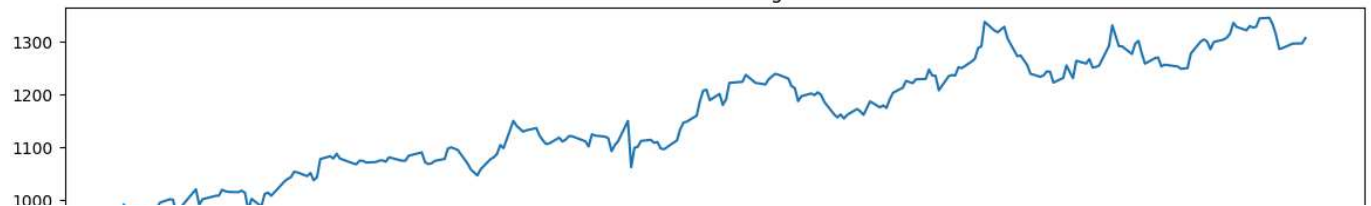
```
plt.figure(figsize=(12, 6))
plt.subplot(2, 1, 1)
hdfc_data['Close'].plot(title="HDFC Bank - Closing Price")
plt.subplot(2, 1, 2)
hdfc_data['Daily Return'].plot(title="HDFC Bank - Daily Returns", color='orange')
plt.tight_layout()
plt.show()
```



```
plt.figure(figsize=(12, 6))
plt.subplot(2, 1, 1)
ic_data['Close'].plot(title="ICICI Bank - Closing Price")
plt.subplot(2, 1, 2)
ic_data['Daily Return'].plot(title="ICICI Bank - Daily Returns", color='orange')
plt.tight_layout()
plt.show()
```



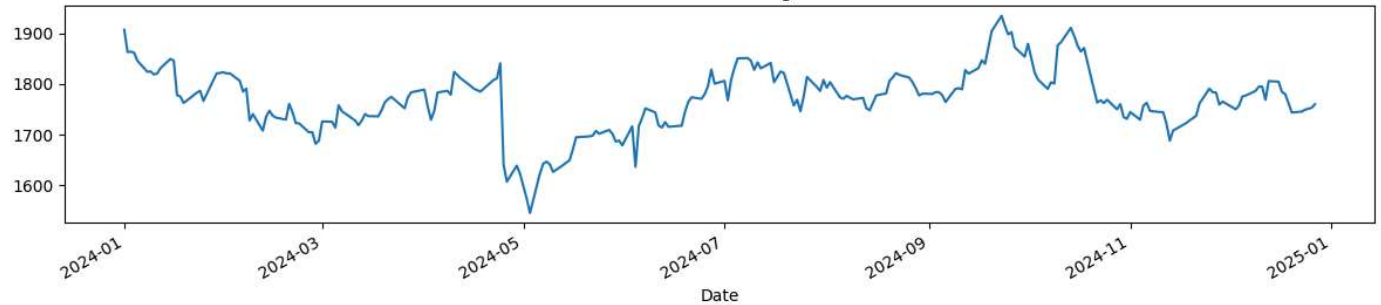
ICICI Bank - Closing Price



```
plt.figure(figsize=(12, 6))
plt.subplot(2, 1, 1)
kb_data['Close'].plot(title="Kotak Bank - Closing Price")
plt.subplot(2, 1, 2)
kb_data['Daily Return'].plot(title="Kotak Bank - Daily Returns", color='orange')
plt.tight_layout()
plt.show()
```



Kotak Bank - Closing Price



Kotak Bank - Daily Returns

