In [10]:
```python
import yfinance as yf
import pandas as pd
```

In [17]:
```python
symbols = ["BTC-USD", "ETH-USD", "BNB-USD"]
```

In [67]:
```python
data = yf.download(symbols, period="7d", interval="1d")
data
```

[*********************100%***********************] 3 of 3 completed

Out[67]:

Price	Close			High			Low		
Ticker	BNB-USD	BTC-USD	ETH-USD	BNB-USD	BTC-USD	ETH-USD	BNB-USD	BTC-USD	ETH-USD
Date									
2025-04-20	592.972412	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.691528	5!
2025-04-21	597.414185	87518.906250	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147461	5!
2025-04-22	618.826965	93441.890625	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000366	5!
2025-04-23	605.815002	93699.109375	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924072	6!
2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740723	6!
2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331909	6!
2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928711	6!

In [68]:
```python
data = data.reset_index()
data
```

Out[68]:

Price	Date	Close						High		ETH-U
Ticker		BNB-USD	BTC-USD	ETH-USD	BNB-USD	BTC-USD	ETH-USD	BNB-USD	BTC-USD	ETH-U
0	2025-04-20	592.972412	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.69
1	2025-04-21	597.414185	87518.906250	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147
2	2025-04-22	618.826965	93441.890625	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000
3	2025-04-23	605.815002	93699.109375	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924
4	2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740
5	2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331
6	2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928

In [70]:
```python
data.columns = ['_'.join(col).strip() if isinstance(col, tuple) else col for col in data.columns]
data
```

Out[70]:

	Date_	Close_BNB-USD	Close_BTC-USD	Close_ETH-USD	High_BNB-USD	High_BTC-USD	High_ETH-USD	Low_BNB-USD	Low_BTC-USD	Low_ETH-USD
0	2025-04-20	592.972412	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.691528
1	2025-04-21	597.414185	87518.906250	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147461
2	2025-04-22	618.826965	93441.890625	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000366
3	2025-04-23	605.815002	93699.109375	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924072
4	2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740723
5	2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331909
6	2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928711

In [71]:
```python
# Rename columns for clarity
data = data.rename(columns={
    "Date": "date",
    "Open_BTC-USD": "open_btc",
    "High_BTC-USD": "high_btc",
    "Low_BTC-USD": "low_btc",
    "Close_BTC-USD": "close_btc",
    "Adj Close_BTC-USD": "adj_close_btc",
    "Volume_BTC-USD": "volume_btc",
    "Open_ETH-USD": "open_eth",
    "High_ETH-USD": "high_eth",
    "Low_ETH-USD": "low_eth",
    "Close_ETH-USD": "close_eth",
    "Adj Close_ETH-USD": "adj_close_eth",
    "Volume_ETH-USD": "volume_eth",
    "Open_BNB-USD": "open_bnb",
    "High_BNB-USD": "high_bnb",
```

```
        "Low_BNB-USD": "low_bnb",
        "Close_BNB-USD": "close_bnb",
        "Adj Close_BNB-USD": "adj_close_bnb",
        "Volume_BNB-USD": "volume_bnb"
})
```

In [72]: `data`

Out[72]:

	Date_	close_bnb	close_btc	close_eth	high_bnb	high_btc	high_eth	low_bnb	low_btc	low_eth
0	2025-04-20	592.972412	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.691528
1	2025-04-21	597.414185	87518.906250	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147461
2	2025-04-22	618.826965	93441.890625	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000366
3	2025-04-23	605.815002	93699.109375	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924072
4	2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740723
5	2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331909
6	2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928711

In [73]:
```
data.loc[1:3, 'close_btc'] = float("nan")
data
```

Out[73]:

	Date_	close_bnb	close_btc	close_eth	high_bnb	high_btc	high_eth	low_bnb	low_btc	low_eth
0	2025-04-20	592.972412	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.691528
1	2025-04-21	597.414185	NaN	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147461
2	2025-04-22	618.826965	NaN	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000366
3	2025-04-23	605.815002	NaN	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924072
4	2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740723
5	2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331909
6	2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928711

In [74]:
```python
data.loc[2, "volume_eth"] = 1e12
data
```

Out[74]:

	Date_	close_bnb	close_btc	close_eth	high_bnb	high_btc	high_eth	low_bnb	low_btc	low_eth
0	2025-04-20	592.972412	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.691528
1	2025-04-21	597.414185	NaN	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147461
2	2025-04-22	618.826965	NaN	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000366
3	2025-04-23	605.815002	NaN	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924072
4	2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740723
5	2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331909
6	2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928711

In [75]:
```python
data.loc[0, "close_bnb"] = 0
data
```

Out[75]:

	Date_	close_bnb	close_btc	close_eth	high_bnb	high_btc	high_eth	low_bnb	low_btc	low_eth
0	2025-04-20	0.000000	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.691528
1	2025-04-21	597.414185	NaN	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147461
2	2025-04-22	618.826965	NaN	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000366
3	2025-04-23	605.815002	NaN	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924072
4	2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740723
5	2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331909
6	2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928711

In [76]:
```python
data.to_csv("crypto_market_data.csv", index=False)
print("Crypto market data saved to csv file")
```

Crypto market data saved to csv file

In [77]:
```python
df = pd.read_csv("crypto_market_data.csv")
df
```

Out[77]:

	Date_	close_bnb	close_btc	close_eth	high_bnb	high_btc	high_eth	low_bnb	low_btc	low_eth
0	2025-04-20	0.000000	85174.304688	1587.514282	595.154114	85306.382812	1618.436768	587.765381	83976.843750	1566.691528
1	2025-04-21	597.414185	NaN	1579.734497	607.820251	88460.093750	1656.119629	592.676147	85143.835938	1566.147461
2	2025-04-22	618.826965	NaN	1757.331665	618.990051	93817.382812	1773.639160	595.880981	87084.531250	1542.000366
3	2025-04-23	605.815002	NaN	1796.104248	619.862488	94535.734375	1829.709961	601.954163	91962.960938	1746.924072
4	2025-04-24	601.715820	93943.796875	1769.831665	606.744324	94016.195312	1801.691406	594.395081	91696.710938	1724.740723
5	2025-04-25	600.181030	94720.500000	1786.634521	610.179626	95768.390625	1826.696045	597.770325	92898.593750	1740.331909
6	2025-04-26	605.976746	94319.929688	1793.933838	605.976746	95211.093750	1831.204102	600.182983	93986.718750	1783.928711

In [78]: `print(f"Dataset shape: {df.shape}")`

Dataset shape: (7, 16)

In [79]: `print(f"Dataset info: {df.info()}")`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7 entries, 0 to 6
Data columns (total 16 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Date_       7 non-null      object
 1   close_bnb   7 non-null      float64
 2   close_btc   4 non-null      float64
 3   close_eth   7 non-null      float64
 4   high_bnb    7 non-null      float64
 5   high_btc    7 non-null      float64
 6   high_eth    7 non-null      float64
 7   low_bnb     7 non-null      float64
 8   low_btc     7 non-null      float64
 9   low_eth     7 non-null      float64
 10  open_bnb    7 non-null      float64
 11  open_btc    7 non-null      float64
 12  open_eth    7 non-null      float64
 13  volume_bnb  7 non-null      int64
 14  volume_btc  7 non-null      int64
 15  volume_eth  7 non-null      int64
dtypes: float64(12), int64(3), object(1)
memory usage: 1.0+ KB
Dataset info: None
```

In [80]:
```python
print(f"Dataset head: {df.head}")
```

```
Dataset head: <bound method NDFrame.head of          Date_    close_bnb    close_btc    close_eth    high_bnb  \
0  2025-04-20     0.000000  85174.304688  1587.514282   595.154114
1  2025-04-21   597.414185           NaN  1579.734497   607.820251
2  2025-04-22   618.826965           NaN  1757.331665   618.990051
3  2025-04-23   605.815002           NaN  1796.104248   619.862488
4  2025-04-24   601.715820  93943.796875  1769.831665   606.744324
5  2025-04-25   600.181030  94720.500000  1786.634521   610.179626
6  2025-04-26   605.976746  94319.929688  1793.933838   605.976746


       high_btc      high_eth      low_bnb       low_btc       low_eth  \
0  85306.382812   1618.436768   587.765381  83976.843750   1566.691528
1  88460.093750   1656.119629   592.676147  85143.835938   1566.147461
2  93817.382812   1773.639160   595.880981  87084.531250   1542.000366
3  94535.734375   1829.709961   601.954163  91962.960938   1746.924072
4  94016.195312   1801.691406   594.395081  91696.710938   1724.740723
5  95768.390625   1826.696045   597.770325  92898.593750   1740.331909
6  95211.093750   1831.204102   600.182983  93986.718750   1783.928711


       open_bnb      open_btc      open_eth    volume_bnb    volume_btc  \
0   591.776794  85066.070312   1612.966064   1307477245   14664050812
1   592.972412  85171.539062   1587.460938   1430343137   41396190190
2   597.417175  87521.875000   1579.815186   1734142941   55899038456
3   618.826965  93427.585938   1757.196533   1834047719   41719568821
4   605.815002  93692.398438   1795.979492   1573943361   31483175315
5   601.715271  93954.250000   1770.003662   1735398071   40915232364
6   600.182983  94708.265625   1786.196167   1558884736   23942998016


       volume_eth
0     7642784469
1    15403785611
2  1000000000000
3    22904644756
4    15207402759
5    17459399281
6    13740555264  >
```

In [81]: `print(f"Dataset describe: {df.describe()}")`

```
Dataset describe:        close_bnb      close_btc      close_eth       high_bnb       high_btc  \
count     7.000000       4.000000       7.000000       7.000000       7.000000
mean    518.561393   92039.632812    1724.440674     609.246800   92445.039062
std     228.767686    4587.859841      97.180173       8.427024    3963.558850
min       0.000000   85174.304688    1579.734497     595.154114   85306.382812
25%     598.797607   91751.423828    1672.422974     606.360535   91138.738281
50%     601.715820   94131.863281    1769.831665     607.820251   94016.195312
75%     605.895874   94420.072266    1790.284180     614.584839   94873.414062
max     618.826965   94720.500000    1796.104248     619.862488   95768.390625

            high_eth       low_bnb        low_btc        low_eth       open_bnb  \
count      7.000000      7.000000       7.000000       7.000000       7.000000
mean    1762.499581    595.803580   89535.742188    1667.252110     601.243801
std       88.590453      4.782960    4038.971965     103.786021       9.166009
min     1618.436768    587.765381   83976.843750    1542.000366     591.776794
25%     1714.879395    593.535614   86114.183594    1566.419495     595.194794
50%     1801.691406    595.880981   91696.710938    1724.740723     600.182983
75%     1828.203003    598.976654   92430.777344    1743.627991     603.765137
max     1831.204102    601.954163   93986.718750    1783.928711     618.826965

            open_btc      open_eth     volume_bnb     volume_btc     volume_eth
count       7.000000      7.000000   7.000000e+00   7.000000e+00   7.000000e+00
mean    90505.997768   1698.516863   1.596320e+09   3.571718e+10   1.560512e+11
std      4381.639236     99.568014   1.860703e+08   1.355482e+10   3.721741e+11
min     85066.070312   1579.815186   1.307477e+09   1.466405e+10   7.642784e+09
25%     86346.707031   1600.213501   1.494614e+09   2.771309e+10   1.447398e+10
50%     93427.585938   1757.196533   1.573943e+09   4.091523e+10   1.540379e+10
75%     93823.324219   1778.099915   1.734771e+09   4.155788e+10   2.018202e+10
max     94708.265625   1795.979492   1.834048e+09   5.589904e+10   1.000000e+12
```

```python
In [82]: print(f"Dataframe tail: {df.tail()}")
```

```
Dataframe tail:        Date_   close_bnb    close_btc    close_eth    high_bnb  \
2  2025-04-22  618.826965         NaN  1757.331665  618.990051
3  2025-04-23  605.815002         NaN  1796.104248  619.862488
4  2025-04-24  601.715820  93943.796875  1769.831665  606.744324
5  2025-04-25  600.181030  94720.500000  1786.634521  610.179626
6  2025-04-26  605.976746  94319.929688  1793.933838  605.976746

       high_btc     high_eth     low_bnb      low_btc      low_eth  \
2  93817.382812  1773.639160  595.880981  87084.531250  1542.000366
3  94535.734375  1829.709961  601.954163  91962.960938  1746.924072
4  94016.195312  1801.691406  594.395081  91696.710938  1724.740723
5  95768.390625  1826.696045  597.770325  92898.593750  1740.331909
6  95211.093750  1831.204102  600.182983  93986.718750  1783.928711

       open_bnb      open_btc     open_eth   volume_bnb   volume_btc  \
2  597.417175  87521.875000  1579.815186  1734142941  55899038456
3  618.826965  93427.585938  1757.196533  1834047719  41719568821
4  605.815002  93692.398438  1795.979492  1573943361  31483175315
5  601.715271  93954.250000  1770.003662  1735398071  40915232364
6  600.182983  94708.265625  1786.196167  1558884736  23942998016

        volume_eth
2  1000000000000
3    22904644756
4    15207402759
5    17459399281
6    13740555264
```

In [88]: `print(f"Data types of each columns: {df.dtypes}")`

```
Data types of each columns: Date_         object
close_bnb     float64
close_btc     float64
close_eth     float64
high_bnb      float64
high_btc      float64
high_eth      float64
low_bnb       float64
low_btc       float64
low_eth       float64
open_bnb      float64
open_btc      float64
open_eth      float64
volume_bnb      int64
volume_btc      int64
volume_eth      int64
dtype: object
```

In [95]:
```python
mean_price = df[["close_btc", "close_eth"]].mean()
mean_price
```

Out[95]:
```
close_btc    92039.632812
close_eth     1724.440674
dtype: float64
```

In [ ]: