Objective: Scrape historical stock data for Nifty500 companies from January 10, 2010, to January 10, 2025, and compute additional financial features.

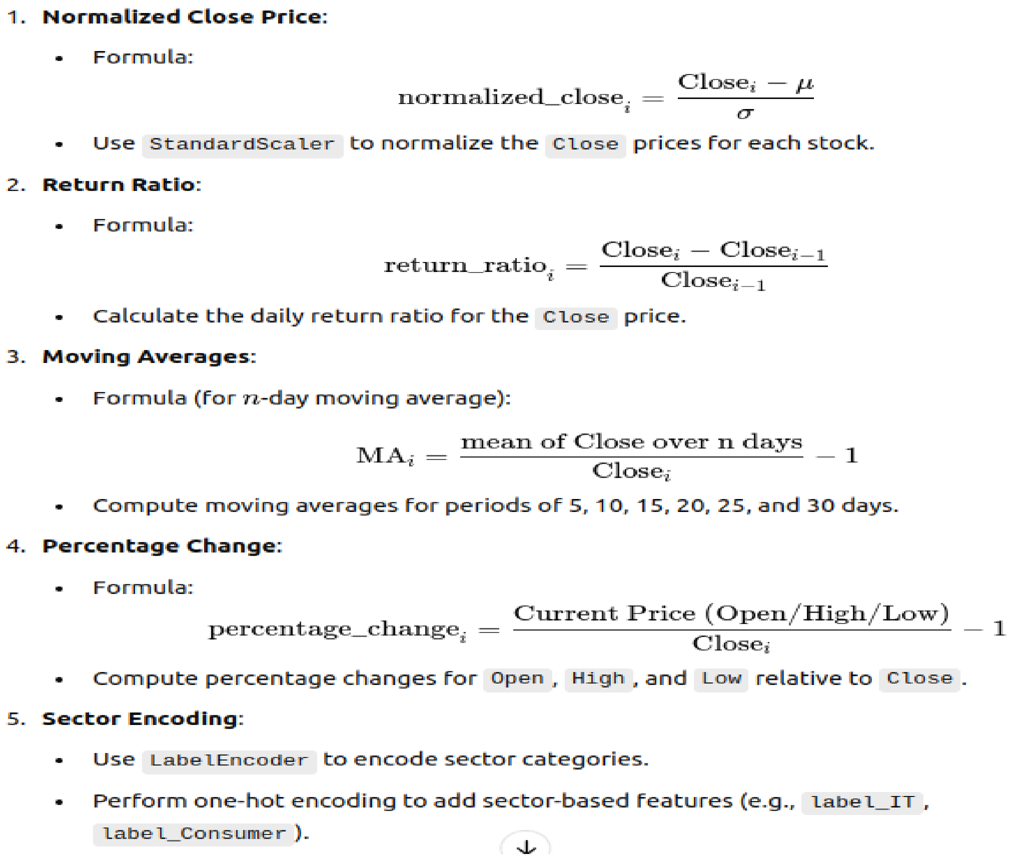
Step 1: Load Company and Sector Information
File: ind_nifty500list.csv
Contains Nifty500 company names and their sector information.
Action:
Load the CSV file
Extract company names and sector information.
Step 2: Scrape Historical Stock Data
Date Range: January 10, 2010, to January 10, 2025 (15 years).
Source: Yahoo Finance (yfinance library).
For each company in the list, download the stock data using the company ticker
Step 3: Check Date Consistency
Objective: Ensure all stocks have data for the same dates.
Extract the reference date range (from one of the stocks, e.g., ONGC).
Filter each company's stock data to match this reference date range.

Sample Code:

```
need_day = np.array(NIFTY50_stock["ONGC"]["stock_price"]['Date'])
for target in NIFTY50_stock.keys():
   if 'Date' in NIFTY50_stock[target]["stock_price"].columns:
                                NIFTY50_stock[target]["stock_price"]    =
NIFTY50_stock[target]["stock_price"][

NIFTY50_stock[target]["stock_price"]['Date'].isin(need_day)].reset_index(d
rop=True)
                           NIFTY50_stock[target]["stock_price"].index    =
NIFTY50_stock[target]["stock_price"]['Date']
   else:
       print(f"'Date' column missing for {target}. Skipping...")
```

Step 4: Compute Additional Features



1. **Normalized Close Price**:
   - Formula:

$$\text{normalized\_close}_i = \frac{\text{Close}_i - \mu}{\sigma}$$

   - Use `StandardScaler` to normalize the `Close` prices for each stock.
2. **Return Ratio**:
   - Formula:

$$\text{return\_ratio}_i = \frac{\text{Close}_i - \text{Close}_{i-1}}{\text{Close}_{i-1}}$$

   - Calculate the daily return ratio for the `Close` price.
3. **Moving Averages**:
   - Formula (for n-day moving average):

$$\text{MA}_i = \frac{\text{mean of Close over n days}}{\text{Close}_i} - 1$$

   - Compute moving averages for periods of 5, 10, 15, 20, 25, and 30 days.
4. **Percentage Change**:
   - Formula:

$$\text{percentage\_change}_i = \frac{\text{Current Price (Open/High/Low)}}{\text{Close}_i} - 1$$

   - Compute percentage changes for `Open`, `High`, and `Low` relative to `Close`.
5. **Sector Encoding**:
   - Use `LabelEncoder` to encode sector categories.
   - Perform one-hot encoding to add sector-based features (e.g., `label_IT`, `label_Consumer`).

Note:

When adding for example the return_ratio to NIFTY50_stock[target]["stock_price"], explicitly assign it as NIFTY50_stock[target]["stock_price"]["return_ratio"] to ensure the new column is directly added to the stock_price DataFrame within the target stock's data structure.

Step 5: Save and Verify

- **Save the processed stock data into a dictionary structure for later use.**

Print summaries to confirm the data structure:

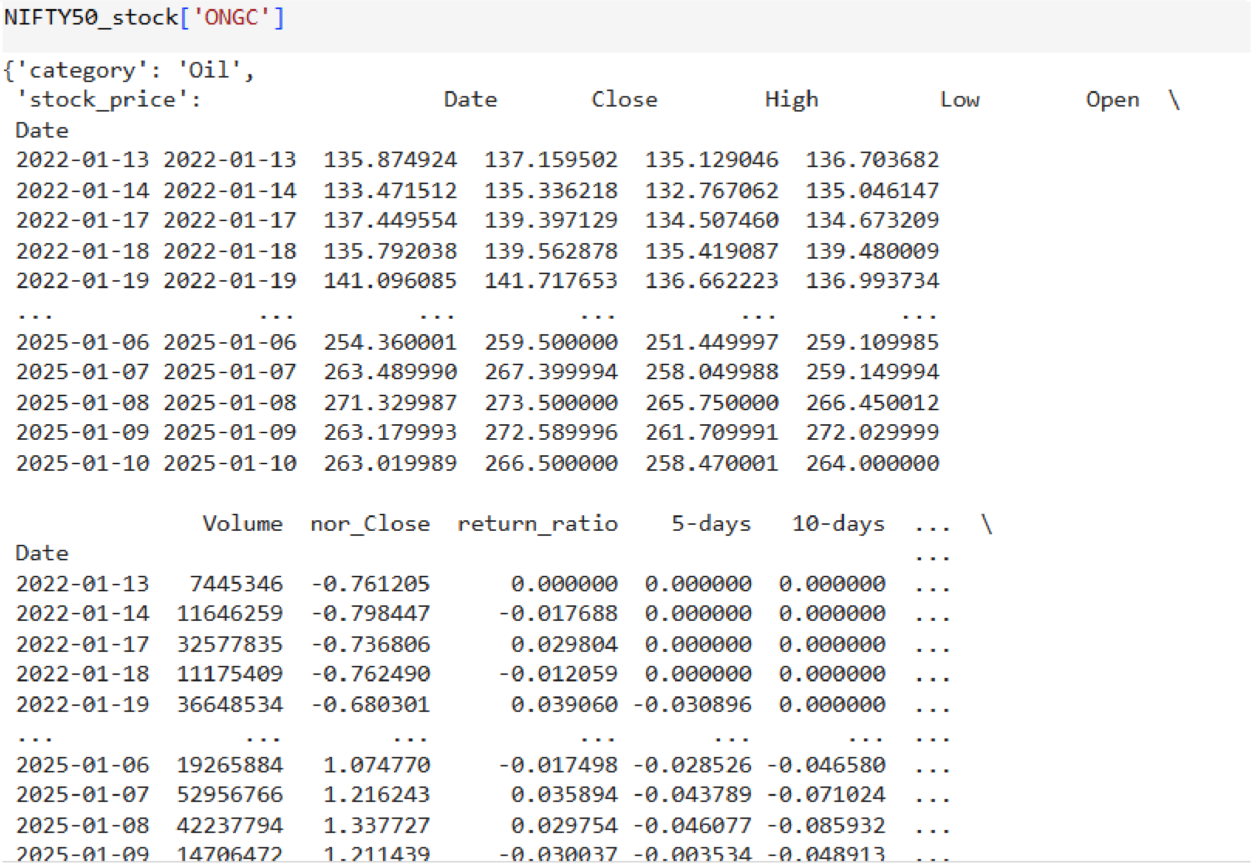
Code

```
print(NIFTY500_stock.keys())  # List of company names
```

```
Sample Output:dict_keys(['COFORGE', 'PERSISTENT', 'TECHM', 'LTIM', 'LTTS', 'TCS', 'INFY',
'HCLTECH', 'WIPRO', 'MPHASIS', 'CENTURYPLY', 'DIXON', 'WHIRLPOOL', 'HAVELLS',
'KALYANKJIL', 'VOLTAS', 'TITAN', 'BATAINDIA', 'RAJESHEXPO', 'AMBER', 'ATGL', 'HINDPETRO',
'PETRONET', 'GAIL', 'OIL', 'ONGC', 'IOC', 'GSPL', 'RELIANCE', 'IGL', 'EICHERMOT',
'APOLLOTYRE', 'BOSCHLTD', 'BALKRISIND', 'TVSMOTOR', 'MRF', 'M&M', 'ASHOKLEY',
```

'BAJAJ-AUTO', 'MARUTI', 'MUTHOOTFIN', 'ICICIGI', 'ICICIBANK', 'HDFCAMC', 'BAJFINANCE', 'SHRIRAMFIN', 'AXISBANK', 'SBICARD', 'HDFCBANK', 'CHOLAFIN'])

```
NIFTY50_stock['ONGC']
```

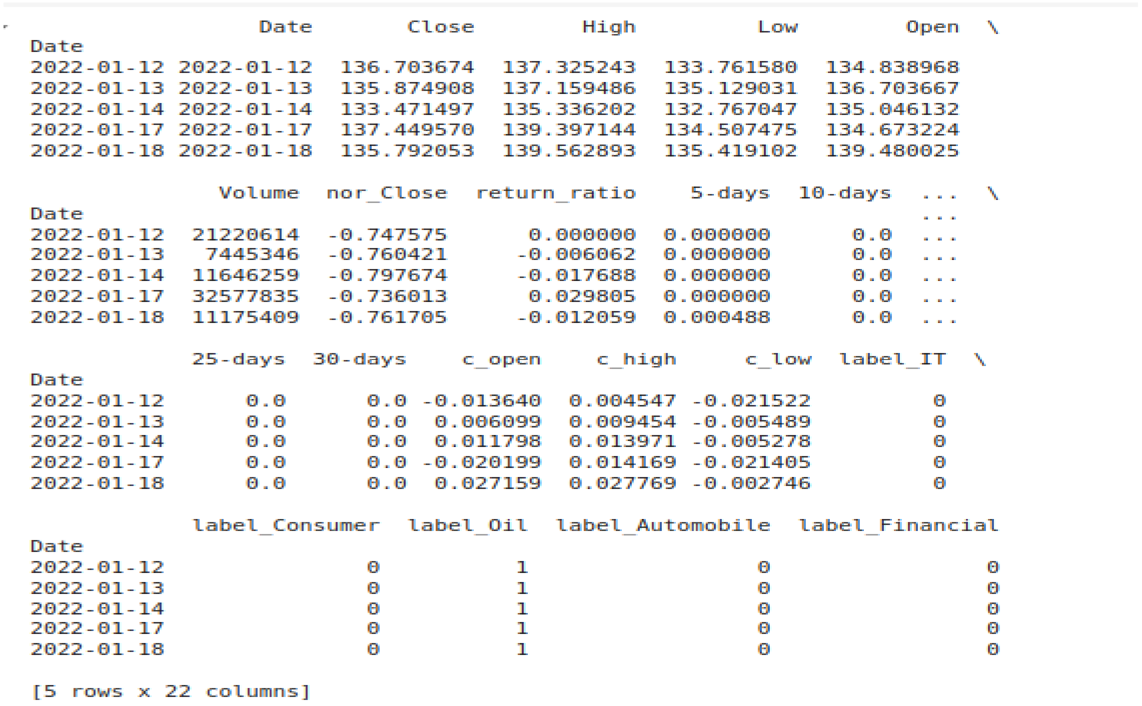


```
NIFTY50_stock['ONGC']
```

```
{'category': 'Oil',
 'stock_price':                  Date       Close        High         Low        Open  \
 Date
 2022-01-13 2022-01-13  135.874924  137.159502  135.129046  136.703682
 2022-01-14 2022-01-14  133.471512  135.336218  132.767062  135.046147
 2022-01-17 2022-01-17  137.449554  139.397129  134.507460  134.673209
 2022-01-18 2022-01-18  135.792038  139.562878  135.419087  139.480009
 2022-01-19 2022-01-19  141.096085  141.717653  136.662223  136.993734
 ...               ...         ...         ...         ...         ...
 2025-01-06 2025-01-06  254.360001  259.500000  251.449997  259.109985
 2025-01-07 2025-01-07  263.489990  267.399994  258.049988  259.149994
 2025-01-08 2025-01-08  271.329987  273.500000  265.750000  266.450012
 2025-01-09 2025-01-09  263.179993  272.589996  261.709991  272.029999
 2025-01-10 2025-01-10  263.019989  266.500000  258.470001  264.000000

               Volume  nor_Close  return_ratio    5-days   10-days  ...  \
 Date                                                                ...
 2022-01-13   7445346  -0.761205      0.000000  0.000000  0.000000  ...
 2022-01-14  11646259  -0.798447     -0.017688  0.000000  0.000000  ...
 2022-01-17  32577835  -0.736806      0.029804  0.000000  0.000000  ...
 2022-01-18  11175409  -0.762490     -0.012059  0.000000  0.000000  ...
 2022-01-19  36648534  -0.680301      0.039060 -0.030896  0.000000  ...
 ...              ...        ...           ...       ...       ...  ...
 2025-01-06  19265884   1.074770     -0.017498 -0.028526 -0.046580  ...
 2025-01-07  52956766   1.216243      0.035894 -0.043789 -0.071024  ...
 2025-01-08  42237794   1.337727      0.029754 -0.046077 -0.085932  ...
 2025-01-09  14706472   1.211439     -0.030037 -0.003534 -0.048913  ...
```

```
NIFTY500_stock['ONGC']['stock_price'].head()  # Sample stock data
```



```
                 Date       Close        High         Low        Open  \
Date
2022-01-12 2022-01-12  136.703674  137.325243  133.761580  134.838968
2022-01-13 2022-01-13  135.874908  137.159486  135.129031  136.703667
2022-01-14 2022-01-14  133.471497  135.336202  132.767047  135.046132
2022-01-17 2022-01-17  137.449570  139.397144  134.507475  134.673224
2022-01-18 2022-01-18  135.792053  139.562893  135.419102  139.480025

              Volume  nor_Close  return_ratio    5-days  10-days  ...  \
Date                                                              ...
2022-01-12  21220614  -0.747575      0.000000  0.000000      0.0  ...
2022-01-13   7445346  -0.760421     -0.006062  0.000000      0.0  ...
2022-01-14  11646259  -0.797674     -0.017688  0.000000      0.0  ...
2022-01-17  32577835  -0.736013      0.029805  0.000000      0.0  ...
2022-01-18  11175409  -0.761705     -0.012059  0.000488      0.0  ...

            25-days  30-days    c_open    c_high     c_low  label_IT  \
Date
2022-01-12      0.0      0.0 -0.013640  0.004547 -0.021522         0
2022-01-13      0.0      0.0  0.006099  0.009454 -0.005489         0
2022-01-14      0.0      0.0  0.011798  0.013971 -0.005278         0
2022-01-17      0.0      0.0 -0.020199  0.014169 -0.021405         0
2022-01-18      0.0      0.0  0.027159  0.027769 -0.002746         0

            label_Consumer  label_Oil  label_Automobile  label_Financial
Date
2022-01-12               0          1                 0                0
2022-01-13               0          1                 0                0
2022-01-14               0          1                 0                0
2022-01-17               0          1                 0                0
2022-01-18               0          1                 0                0

[5 rows x 22 columns]
```