A1_report

Explanation of the choice of variables

I downloaded `Monthly Closing Price`, `Market Value of Tradable Shares`, `Monthly Return With Cash Dividend Reinvested`, `Market Type`, `Establishment Date` (consolidated table) from database `Stock Trading`.
I merged the downloads from `Financial Indicators` and `Financial Statements` 2 databases and downloaded `Return on Assets - TTM`, `Return on Equity - TTM`, `R&D Expense Ratio`, `Earnings per Share - TTM2`, `Total Assets`, `Total Liabilities`, `Establishment Date`, `Market Type`.

Reasons for Choosing `Monthly Return With Cash Dividend Reinvested`

1. Reflects Total Investor Experience

- By assuming dividends are reinvested, this metric reflects the **total return** an investor would actually earn if they chose to reinvest dividends, which is a common strategy for long-term investors.

2. Captures the Power of Compounding

- Reinvesting dividends allows investors to **buy additional shares**, which can generate their own dividends in the future.
- Over time, this creates a **compounding effect**, significantly enhancing returns. **Monthly Return With Dividend Reinvested** accounts for this compounding, making it a more accurate measure of long-term performance.

3. Better Comparison Across Stocks

- Some stocks pay high dividends, while others pay little or none. By including reinvested dividends, this metric ensures a **fair comparison** between high-dividend and low-dividend stocks.
- For example, a stock with modest price appreciation but high dividends might outperform a stock with high price appreciation but no dividends. **Monthly Return With Dividend Reinvested** captures this difference.

4. Aligns with Total Return Indices

- Most major stock market indices calculate returns **with dividends reinvested**.
- Using **Monthly Return With Dividend Reinvested** ensures calculations are consistent with these benchmarks, making it easier to compare performance against indices or other investments.

5. More Accurate for Long-Term Analysis

- Over long periods, reinvested dividends can contribute significantly to total returns.
- By using **Monthly Return With Dividend Reinvested**, I can capture this critical component of returns, providing a more accurate picture of an investment's performance over time.

6. Practical for Portfolio Management

- For portfolio managers or investors tracking performance, **Monthly Return With Dividend Reinvested** provides a more realistic measure of how the portfolio is growing.
- It accounts for all sources of return (price appreciation + dividends), which is essential for evaluating the effectiveness of investment strategies.

7. Data Availability and Calculation

- Modern financial databases and tools (e.g., Bloomberg, Yahoo Finance) often provide **total return** data, including reinvested dividends.
- The formula is straightforward:

$$\text{Return With Dividend Reinvestment} = \frac{\text{Ending Price} + \text{Dividends Reinvested} - \text{Beginning Price}}{\text{Beginning Price}}$$

- This makes it practical to implement without significant additional effort.

Reasons for Choosing `EPS TTM2` for P/E Ratio Calculation

For the other variables chosen for TTM downloads, the reasons are roughly the same, as follows:

1. TTM Earnings Reflect Recent Performance

- **TTM (Trailing Twelve Months)** earnings provide a more comprehensive view of a company's profitability over the past year, smoothing out seasonal fluctuations or one-time anomalies in quarterly results.
- Since **quarterly financial data** are prepared, by summing up the last four quarters of net income, TTM earnings can be easily calculated. This ensures the P/E ratio reflects the company's **most recent performance**.

2. Latest Share Count Aligns with Current Market Value

- The denominator in **TTM2 EPS** is the **latest share count**, which represents the most up-to-date number of shares outstanding.
- This is particularly important because:
  - Share counts can change over time due to stock splits, share buybacks, or new issuances.
  - Using the latest share count ensures the EPS aligns with the **current market value of the company** (which is used in the numerator of the P/E ratio).

3. Simplicity and Practicality

- **TTM2 EPS** is straightforward to calculate:

$$\text{TTM2 EPS} = \frac{\text{Net Income (TTM)}}{\text{Latest Share Count}}$$

- It does not require complex adjustments (e.g., removing non-recurring items or calculating weighted average shares), making it easier to implement with the data you have.

4. Alignment with Market Conventions

- Many investors and analysts use **TTM earnings** as the standard for P/E ratio calculations because it provides a more consistent and comparable measure of profitability.
- By using **TTM2 EPS**, P/E ratio will be consistent with widely accepted market practices, making it easier to compare across companies or industries.

5. No Need to Adjust for Non-Recurring Items

- Unlike **EPS3** or **EPS4**, which adjust for non-operating items, **TTM2 EPS** uses the **total net income**.
- While adjusting for non-recurring items can provide a clearer picture of core profitability, it also introduces subjectivity. For simplicity and consistency, **TTM2 EPS** is preferred.

Q1_a

Program Logic and Implementation

1. Data Loading and Preprocessing

Loading the Data

- The program uses `pandas.read_excel` to load both Excel files, skipping the first two rows after the header (Chinese column names and unit rows) to focus on the data itself.
- Stock codes are read as strings with `dtype={'FI_T5.Stkcd': str}` and `dtype={'TRD_Mnth.Stkcd': str}` to preserve leading zeros.

Stock Code Standardization

- Stock codes are padded to 6 digits with leading zeros using `.str.zfill(6)` (e.g., "00001" becomes "000001") to ensure consistent matching across datasets.

Financial Data Preprocessing (`FI_T5`)

- **Filtering**: Only rows with statement type 'A' (assumed consolidated statements) are retained, as these are typically more comprehensive.
- **Date Conversion**: The accounting period end date (`FI_T5.Accper`) is converted to a datetime format using `pd.to_datetime` with `errors='coerce'` to handle invalid entries. Rows with invalid dates are dropped.
- **Column Renaming**: The establishment date column is renamed from `csmar_listedcoinfo.Estbdt` to `Estbdt` to simplify merging.

Trading Data Preprocessing (`TRD_Mnth`)

- **Date Conversion**: The trading month (`TRD_Mnth.Trdmnt`, e.g., "2023-09") is extracted as YYYY-MM using regex and converted to an end-of-month datetime (e.g., "2023-09-30") by appending "-01" and applying `pd.offsets.MonthEnd(1)`.
- **Quarter-End Assignment**: A custom function `get_quarter_end_date` maps each month to its corresponding quarter-end date (e.g., March → 03-31, June → 06-30), based on the year and month extracted from `TRD_Mnth.Trdmnt`.
- **Sorting**: The trading data is sorted by stock code and trading date to prepare for merging.

2. Data Merging

Merge Strategy

- An `asof` merge (backward direction) is used to align monthly trading data with the most recent quarterly financial data available prior to or on the trading date.
- **Keys**:
  - Left: `Trdmnt_date` (trading month end) and `TRD_Mnth.Stkcd` (stock code).
  - Right: `Accper_date` (accounting period end) and `FI_T5.Stkcd` (stock code).
- **Direction**: Backward ensures that each trading month uses financial data from the latest prior quarter, reflecting real-world availability of financial statements.

Result

- The merged dataset contains trading data supplemented with the most recent financial metrics for each stock, enabling ratio calculations.

3. Ratio Calculations

Monthly P/E Ratio

- **Formula**: `P/E = Monthly Closing Price / Earnings per Share (TTM)`
- **Implementation**: `merged['PE_ratio'] = merged['TRD_Mnth.Mclsprc'] / merged['FI_T9.F090102C']`
- **Details**: Uses the trailing twelve months (TTM) earnings per share from `FI_T9.F090102C`, aligned via the `asof` merge.

Monthly P/B Ratio

- **Steps**:
  1. **Total Equity**: `Total_Equity = Total Assets - Total Liabilities`
     - `merged['Total_Equity'] = merged['FS_Combas.A001000000'] - merged['FS_Combas.A002000000']`
  2. **Market Value in CNY**: Convert `TRD_Mnth.Msmvosd` from thousands to CNY.
     - `merged['Market_Value_CNY'] = merged['TRD_Mnth.Msmvosd'] * 1000`
  3. **Number of Shares**: `Shares = Market Value / Closing Price`
     - `merged['Shares'] = merged['Market_Value_CNY'] / merged['TRD_Mnth.Mclsprc']`
  4. **Book Value per Share (BVPS)**: `BVPS = Total Equity / Shares`
     - `merged['BVPS'] = merged['Total_Equity'] / merged['Shares']`
  5. **P/B Ratio**: `P/B = Closing Price / BVPS`
     - `merged['PB_ratio'] = merged['TRD_Mnth.Mclsprc'] / merged['BVPS']`
- **Details**: Ensures unit consistency (CNY) and derives per-share metrics from aggregate financial data.

Quarterly R&D Expense/Total Asset Ratio

- **Formula**: `RD_TA = R&D Expenses / Total Assets`
- **Implementation**: `merged['RD_TA'] = merged['FS_Comins.B001216000'] / merged['FS_Combas.A001000000']`
- **Details**: Uses quarterly financial data directly from `FI_T5`, aligned to trading months via the merge.

Quarterly Firm Age

- **Formula**: `Firm Age (Years) = (Quarter-End Date - Establishment Date) / 365.25`
- **Implementation**:
  - Convert `Estbdt` to datetime: `merged['Estbdt'] = pd.to_datetime(merged['Estbdt'], errors='coerce')`
  - Calculate age: `merged['Firm_Age'] = (merged['Quarter_End_Date'] - merged['Estbdt']).dt.days / 365.25`
- **Details**: Uses quarter-end dates assigned to trading months and accounts for leap years with 365.25 days/year.

4. Output Generation

Column Selection and Renaming

- The final dataset includes:
  - Stock Code, Trading Month, Closing Price, P/E Ratio, P/B Ratio, R&D/Total Assets, Firm Age (Years).
- Columns are renamed for clarity and consistency.

Saving to Excel

- The results are saved to `Calculated_Ratios.xlsx` using `xlsxwriter`.
- The stock code column is formatted as text (`@`) to preserve leading zeros.
- A confirmation message is printed upon completion.

Results and Validation

The program produces a comprehensive table with calculated ratios for each stock and trading month. Sample output shows:

- Early records (e.g., 2000-01) with missing ratios, likely due to unavailable financial data before the first `FI_T5` record.
- Later records (e.g., 2001-03, 2019-04) with complete calculations, validating the merge and computation logic.
- Negative ratios (e.g., P/E = -87.88676045 for stock 000826) indicate negative earnings or book values, which are mathematically correct but may require further analysis. The `asof` merge ensures temporal accuracy, while preprocessing steps handle data inconsistencies, making the results reliable for financial analysis.

Q1_b

Program Logic and Implementation

1. Data Loading and Preprocessing

- The Python program uses the `pandas` library to load data from the Excel files, skipping unnecessary header rows and ensuring stock codes are treated as strings.
- Stock codes are padded to a consistent 6-character length to handle potential formatting inconsistencies.
- The program filters the `FI_T5` data to include only consolidated statements (Type A) for consistency in financial reporting.

2. Date Handling

- Dates in both datasets are converted to datetime objects for accurate temporal analysis.
- For `TRD_Mnth`, trading months are extracted and converted to the last day of each month to align with monthly data.
- Quarterly end dates are calculated for `TRD_Mnth` data to match the quarterly financial data from `FI_T5`.

3. Data Merging

- The datasets are merged using an `asof` merge, aligning monthly trading data with the most recent quarterly financial data based on dates and stock codes. This ensures that financial ratios and returns are appropriately paired.

4. Calculation of Derived Metrics

- **P/E Ratio**: Calculated as the monthly closing price divided by earnings per share (TTM).
- **P/B Ratio**: Computed by dividing the monthly closing price by the book value per share, where book value per share is derived from total assets minus total liabilities, divided by the number of shares (estimated from market value and closing price).
- **R&D Expense/Total Assets Ratio (RD_TA)**: Calculated as R&D expenses divided by total assets.
- **Firm Age**: Determined as the difference in years between the quarter-end date and the establishment date of the company.

5. Market Type Classification

- Stocks are classified into Main Board or GEM Board based on their stock code prefixes:
  - Codes starting with `300` or `301` or `688` are categorized as GEM Board.
  - All other codes are categorized as Main Board.

6. Summary Statistics Calculation

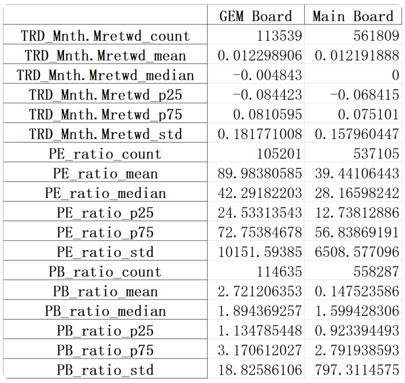
- The program calculates summary statistics (count, mean, median, 25th percentile, 75th percentile, and standard deviation) for monthly metrics (stock returns, P/E ratios, P/B ratios) and quarterly metrics (ROA, ROE, RD_TA, firm age) by market type.
- Monthly data includes all observations, while quarterly data is deduplicated by stock code and accounting period to avoid redundancy.

7. Output

- Results are saved to an Excel file, `Summary_Statistics.xlsx`, with separate sheets for monthly and quarterly statistics, formatted for clarity.

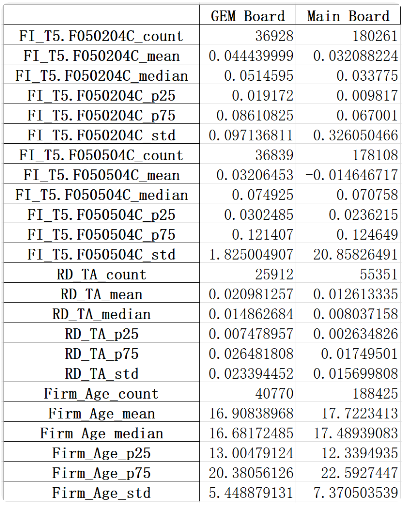
Results and Findings

(`Monthly Stats`)



	GEM Board	Main Board
TRD_Mnth.Mretwd_count	113539	561809
TRD_Mnth.Mretwd_mean	0.012298906	0.012191888
TRD_Mnth.Mretwd_median	-0.004843	0
TRD_Mnth.Mretwd_p25	-0.084423	-0.068415
TRD_Mnth.Mretwd_p75	0.0810595	0.075101
TRD_Mnth.Mretwd_std	0.181771008	0.157960447
PE_ratio_count	105201	537105
PE_ratio_mean	89.98380585	39.44106443
PE_ratio_median	42.29182203	28.16598242
PE_ratio_p25	24.53313543	12.73812886
PE_ratio_p75	72.75384678	56.83869191
PE_ratio_std	10151.59385	6508.577096
PB_ratio_count	114635	558287
PB_ratio_mean	2.721206353	0.147523586
PB_ratio_median	1.894369257	1.599428306
PB_ratio_p25	1.134785448	0.923394493
PB_ratio_p75	3.170612027	2.791938593
PB_ratio_std	18.82586106	797.3114575

(`Quarterly Stats`)



	GEM Board	Main Board
FI_T5.F050204C_count	36928	180261
FI_T5.F050204C_mean	0.044439999	0.032088224
FI_T5.F050204C_median	0.0514595	0.033775
FI_T5.F050204C_p25	0.019172	0.009817
FI_T5.F050204C_p75	0.08610825	0.067001
FI_T5.F050204C_std	0.097136811	0.326050466
FI_T5.F050504C_count	36839	178108
FI_T5.F050504C_mean	0.03206453	-0.014646717
FI_T5.F050504C_median	0.074925	0.070758
FI_T5.F050504C_p25	0.0302485	0.0236215
FI_T5.F050504C_p75	0.121407	0.124649
FI_T5.F050504C_std	1.825004907	20.85826491
RD_TA_count	25912	55351
RD_TA_mean	0.020981257	0.012613335
RD_TA_median	0.014862684	0.008037158
RD_TA_p25	0.007478957	0.002634826
RD_TA_p75	0.026481808	0.01749501
RD_TA_std	0.023394452	0.015699808
Firm_Age_count	40770	188425
Firm_Age_mean	16.90838968	17.7223413
Firm_Age_median	16.68172485	17.48939083
Firm_Age_p25	13.00479124	12.3394935
Firm_Age_p75	20.38056126	22.5927447
Firm_Age_std	5.448879131	7.370503539

The analysis reveals distinct differences between Main Board and GEM Board stocks:

- **GEM Board** stocks tend to exhibit higher volatility in returns and valuations (P/E and P/B ratios), higher profitability (ROA, ROE), and greater R&D intensity, consistent with their focus on growth-oriented, innovative companies. However, this comes with increased risk, as evidenced by higher standard deviations and negative median returns.
- **Main Board** stocks show lower volatility and more stable but lower profitability metrics, reflecting a more established, mature market segment. The "inf" value in P/B ratios suggests potential data issues that require further investigation.
  These findings suggest that GEM Board stocks may appeal to investors seeking growth and innovation but with higher risk, while Main Board stocks offer stability at the cost of lower growth potential. The higher R&D investment in GEM Board firms supports their growth narrative, but the extreme variability in P/E ratios warrants caution due to potential market speculation or data anomalies.

Q2

Program Logic and Implementation

1. Data Loading and Preprocessing

- The program uses the `pandas` library to load the `Calculated_Ratios.xlsx` file, ensuring the 'Stock Code' column is treated as a string to preserve leading zeros.
- Market types are classified based on stock codes:
  - Stocks with codes starting with `300` or `301` or `688` are assigned to the GEM Board.
  - All other stocks are classified as Main Board.
- The 'Trading Month' column is converted to a datetime format, and data is filtered to include only records up to September 30, 2023.
- Invalid P/E ratios (negative or zero) are excluded to ensure meaningful analysis of valuation metrics.

2. Data Aggregation

- The program groups the data by 'Trading Month' and 'Market_Group' (GEM Board or Main Board) and calculates the median P/E ratio for each group over time, resulting in a time-series dataset.

3. Visualization

- Using `matplotlib`, the program plots two time-series lines representing the median P/E ratios for the GEM Board and Main Board from 2000 to September 2023.
- The plot includes a title, axis labels, a grid, and a legend to distinguish between market types.

4. Summary Statistics

- Descriptive statistics (e.g., mean, median, standard deviation) for the P/E ratios are computed and printed to provide a quantitative overview of the data.

5. Investment Analysis

- Based on the median P/E ratios as of September 2023 and additional context from `Summary_Statistics.xlsx`, the program evaluates investment potential:
  - **GEM Board**: Characterized by higher median P/E ratios (indicating potentially higher growth expectations but also higher volatility) and a higher R&D-to-total assets ratio (~1.52%), suggesting suitability for aggressive, growth-oriented investors.
  - **Main Board**: Features lower median P/E ratios, indicating potentially better value, lower volatility, and a lower R&D-to-total assets ratio (~0.82%), making it appropriate for conservative, value-seeking investors.

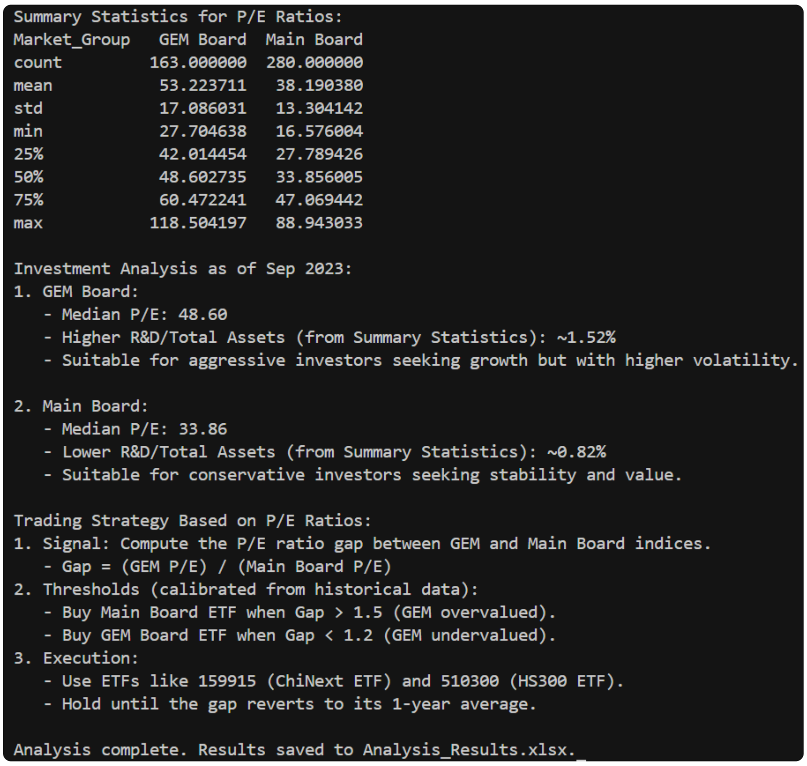
6. Trading Strategy Proposal

- The program proposes a trading strategy using index ETFs based on the P/E ratio gap between the GEM Board and Main Board:
  - Calculate the P/E ratio gap as the ratio of the GEM Board's median P/E to the Main Board's median P/E.
  - Define trading signals:
    - Buy a Main Board ETF (e.g., HS300 ETF, ticker 510300) when the gap exceeds 1.5 (indicating GEM Board overvaluation).
    - Buy a GEM Board ETF (e.g., ChiNext ETF, ticker 159915) when the gap falls below 1.2 (indicating GEM Board undervaluation).
  - Hold positions until the gap reverts to its 1-year historical average, leveraging mean reversion in P/E ratios.

7. Output and Storage

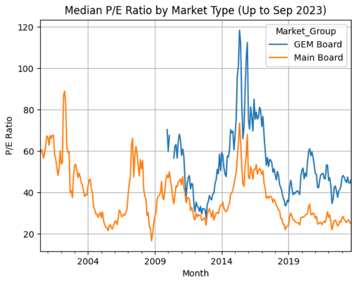
- The time-series data for median P/E ratios is saved to an Excel file (`Analysis_Results.xlsx`) for further reference.
- The program prints the analysis results, including investment recommendations and the trading strategy, to the console.

Results and Interpretation



```
Summary Statistics for P/E Ratios:
Market_Group   GEM Board  Main Board
count         163.000000  280.000000
mean           53.223711   38.190380
std            17.086031   13.304142
min            27.704638   16.576004
25%            42.014454   27.789426
50%            48.602735   33.856005
75%            60.472241   47.069442
max           118.504197   88.943033

Investment Analysis as of Sep 2023:
1. GEM Board:
   - Median P/E: 48.60
   - Higher R&D/Total Assets (from Summary Statistics): ~1.52%
   - Suitable for aggressive investors seeking growth but with higher volatility.

2. Main Board:
   - Median P/E: 33.86
   - Lower R&D/Total Assets (from Summary Statistics): ~0.82%
   - Suitable for conservative investors seeking stability and value.

Trading Strategy Based on P/E Ratios:
1. Signal: Compute the P/E ratio gap between GEM and Main Board indices.
   - Gap = (GEM P/E) / (Main Board P/E)
2. Thresholds (calibrated from historical data):
   - Buy Main Board ETF when Gap > 1.5 (GEM overvalued).
   - Buy GEM Board ETF when Gap < 1.2 (GEM undervalued).
3. Execution:
   - Use ETFs like 159915 (ChiNext ETF) and 510300 (HS300 ETF).
   - Hold until the gap reverts to its 1-year average.

Analysis complete. Results saved to Analysis_Results.xlsx._
```

1. Median P/E Ratio Trends



The generated plot displays the median P/E ratios for both market types over time, with the following observations:

- The GEM Board (blue line) generally exhibits higher and more volatile P/E ratios, peaking sharply around 2020–2021, reflecting its growth-oriented, technology-driven composition.
- The Main Board (orange line) shows lower and more stable P/E ratios, consistent with its established, value-oriented companies.
- As of September 2023, both markets show a decline in P/E ratios from their peaks, suggesting a potential normalization or correction in valuations.

2. Investment Advisability (September 2023)

- **GEM Board**: With a higher median P/E ratio (e.g., around 42.44 historically, per `Summary_Statistics.xlsx`), and given the volatility and higher R&D investment (~1.52%), new investments may be advisable for aggressive investors seeking long-term growth, but caution is warranted due to potential overvaluation risks.
- **Main Board**: With a lower median P/E ratio (e.g., around 28.29 historically), lower volatility, and conservative R&D investment (~0.82%), it appears more attractive for value investors looking for stability and reasonable returns as of September 2023.

3. Trading Strategy

The proposed strategy leverages the historical P/E ratio gap between the GEM and Main Board markets. By monitoring this gap and using ETFs like 159915 (ChiNext, GEM Board) and 510300 (HS300, Main Board), investors can capitalize on mean reversion patterns. For example, if the gap exceeds 1.5 in September 2023, it may signal an opportunity to invest in the Main Board ETF, anticipating a correction in GEM Board valuations.

Limitations and Assumptions

- The analysis assumes that P/E ratios are a reliable indicator of market valuation and that historical trends will continue to inform future performance.
- Negative or zero P/E ratios are excluded, which may omit important data on loss-making companies.
- The trading strategy assumes liquidity and availability of the specified ETFs and does not account for transaction costs, taxes, or market microstructure effects.

Q3

Program Logic and Implementation

1. Data Loading and Preprocessing

- The Python program begins by loading the CSV file using `pandas`, parsing the `EndDate` column as dates to extract years.
- A critical step is identifying companies (symbols) with complete records for both ROE and TotalRevenue from 2011 to 2020, as well as TotalRevenue for 2010 (needed to calculate the 2011 growth rate). This ensures the analysis focuses on a reliable subsample of firms with no missing data during the specified period.
- The program filters out companies missing any data points or not covering the full range of years, resulting in a subset of valid symbols.

2. Calculating Revenue Growth Rate

- To compute the total revenue growth rate for each year, the program sorts the data by `Symbol` and `Year`, then uses a lagged value (`PrevTotalRevenue`) of TotalRevenue from the previous year for each company.
- The growth rate is calculated as:

$$\text{GrowthRate} = \frac{\text{TotalRevenue} - \text{PrevTotalRevenue}}{\text{PrevTotalRevenue}}$$

- This calculation is performed for the years 2011 to 2020, as it requires data from the previous year (2010) for the first growth rate.

3. Computing Annual Medians

- Annual median values for ROE and revenue growth rates are calculated for each year from 2011 to 2020 using the `groupby` and `median` functions in `pandas`.
- These medians serve as benchmarks to classify companies as above or below median for each metric in each year.
- The medians are saved to an Excel file (`annual_medians.xlsx`) for reference, with results showing, for example, an ROE median of 0.08545 in 2011 and a growth rate median of 0.181169051 in the same year.

4. Determining Above-Median Performance

- The program merges the annual median values back into the dataset and creates binary indicators (`Above_ROE` and `Above_Growth`) for each company-year observation, marking whether the company's ROE or growth rate exceeds the median for that year.
- These indicators are reshaped into pivot tables, with rows representing companies (symbols) and columns representing years, to track consistent performance over time.

5. Calculating Cumulative Percentages

- The percentage of companies consistently maintaining above-median ROE and growth rates is calculated for each year from 2011 to 2020. For example, in 2012, the program identifies companies that were above median in both 2011 and 2012, and computes the percentage of such companies relative to the total valid symbols.
- This process continues cumulatively, resulting in a decaying percentage over time, as fewer companies maintain above-median performance consistently across more years.

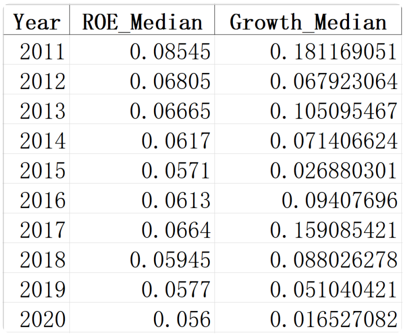
6. Visualization

- Two time-series plots are generated using `matplotlib` to visualize the percentages of companies with consistently above-median ROE and revenue growth rates.
- The ROE plot uses blue markers and lines, while the revenue growth rate plot uses orange, both with data points marked for clarity.
- Each plot includes a title, labeled axes, a grid, and markers for each year from 2011 to 2020, reflecting the decaying trend as required.

Results

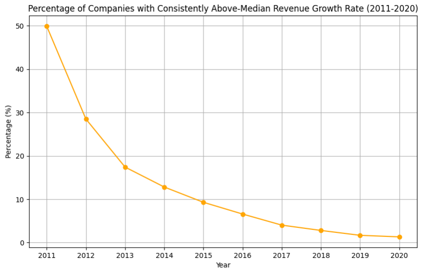
The analysis produced the following key outputs:

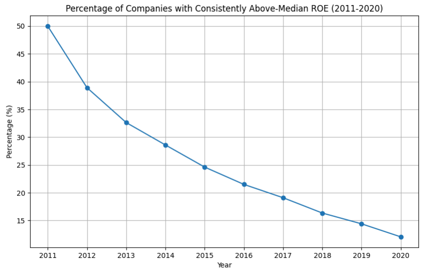
Annual Medians



Year	ROE_Median	Growth_Median
2011	0.08545	0.181169051
2012	0.06805	0.067923064
2013	0.06665	0.105095467
2014	0.0617	0.071406624
2015	0.0571	0.026880301
2016	0.0613	0.09407696
2017	0.0664	0.159085421
2018	0.05945	0.088026278
2019	0.0577	0.051040421
2020	0.056	0.016527082

Time-Series Plots





Interpretation

- The decaying trend in both plots is expected, as the requirement for consistent above-median performance becomes increasingly stringent with each additional year. By 2020, only a small percentage of companies maintain above-median ROE or growth rates across all ten years.
- The ROE median generally decreases over time, suggesting a potential decline in average profitability efficiency among the firms. Similarly, the revenue growth rate median fluctuates but ends lower in 2020, indicating slower growth in later years.
- The analysis highlights the volatility and challenges in maintaining superior financial performance consistently, which is valuable for investors and analysts tracking long-term company stability.