This process and the attached files (if ran) should net the following result. I also attached some data files in python (.pickle) if you do not want to rerun everything. From start to end on my computer it takes about ~40 minutes to run right now. I tried to follow the steps of the Black Jensen Scholes 1972 study, summarized by this screenshot from a PowerPoint somewhere on the internet.   
A white text on a white background

Description automatically generated

1. Sourcing Stock Data *advanced\_fetch\_stock\_data(start\_year, end\_year, n\_stocks) in advanced\_fetch\_stock\_data.py*: Pulls the top 500 stocks by average market cap every single year from start year to end year (inclusive.) This is what it returns:   
   A screenshot of a computer screen

   Description automatically generated

Also getting an array of daily risk-free rate data and value weighted returns (both indexed with dates.) For simplicity I downloaded them one time from CRSP and saved them to excel documents.

1. Calculating the Indexes for Calendar and Event Months:

**From here on out, unless there is a major typo/bug, both should be treated identically. I believe I always had calendar months always as “range 1” and event months always as “range 2.”**

1. Calculating calendar and event month returns.
2. Calculate betas and portfolio returns. Now everything is organized by sequence #. Sequence 0 starts with January 1, 1990, for both. Sequence 60 starts on January 1, 1995 for calendar months and on August 4th, 1997 for event months. Therefore, the total returns will be different.
   1. Form 2 empty portfolios that are represented as 2d arrays that are 10 wide and (number of months – 60)/12 = 25 long.
   2. **For both:** Loop through sequences from 72, 84, 96, …360: (Call it N)
      1. Take the monthly returns from N-72 to N sequence. Filter out any stock that does not have all 72 data points. (This introduces multiple levels of survivorship bias)
      2. Calculate beta for each stock from points 1 to 60 and returns from points 61 to 72. Take the market cap at point 61, which if the calculate monthly returns is correct will be the first day of point 61.
         1. *It should be possible to sum the log returns. For some reason on my code that caused issues so I converted them back to simple, did a (1+return) product, and convert them back to log and got the number that we would expect.*
         2. *I divided the market caps by the sum because numpy was throwing an error because the sum of the weights was too large?*
      3. Rank the stocks by beta into ten portfolios. Note this is not market cap weighted, meaning eventually one portfolio could be worth $100 billion and another be worth $1 Trillion.
      4. For each subportfolio, calculate and append the market weighted average return from points 61 to 72.
      5. Calculate the market return from points 61 to 72 for the SP500 and store it.
   3. From here, calculate the beta of each subportfolio against the corresponding market returns as well as the mean return (forced to 0 intercept as CAPM theory would suggest.) Then, do one last regression not forced to 0 intercept of the betas as the x axis and the subportfolio mean yearly returns as the y axis to get a slope.   
      A graph with red and blue dots

      Description automatically generated