SECTION 6: METHODS AND FUNCTIONS, 2 hours 54 mins, 30 parts

- **6/29 Tuple Unpacking with Python Functions**
  - **defining functions and returning multiple results in Python**
  - **in the .ipynb file**
    - **he's defined an array of tuples -> [(x,y),(,),(,)] <- essentially coordinates inside an array**

    - **for item in stock_prices:**
      - print(item) <- printing each of the tuples by iterating through the list

    - **another example**
      - **for ticker, price in stock_prices: <- this is (x,y)**

    - **another example**
      - we have tuples of x and y
      - -> we want the y with the maximal x

      - **-> def function_name(array_name):**
        - #she's initialising the values to return
        - current_max = 0
        - employee_of_month = ''
        - #then we're iterating through the array of tuples and updating the current_max -> and otherwise it's a pass
        - #return
        - return(employee_of_the_month, current_max)
      - -> then she does an example and it prints the maximum
      - -> **if the function outputs two things (e.g return x,y - then you can call the function and write x,y=function_call())**

    - **if you are reusing someone elses' function**
      - -> e.g if it takes tuples then there can 'not be enough values to unpack' (it will tell you)
      - -> in this case, you can set everything equal to a variable, and then inspect the variable