**In class exercise**

**Takes a list of words and returns the length of the longest one**

* **for programming this you may want to follow this approach,**

**1. having two list one for your words like : ["PHP", "Exercises", "Backend"]**

**2. one for keeping the count of the letters in each word like word\_len**

**3. then iterate through the word\_list, then append the LEN of each word to the list**

**4. you will have a list which contains both the word and len of the word like this [(3,”php),(9,”Exercises”), (7,”backend”)].**

**5. Then sort this list**

**6. then return the last tuple**

Word\_len.append(len(words)): Used append to append an object at end

Word\_freq.extend([(len(words),words)]): Used extend to extend list by appending elements from the iterable

![A screenshot of a social media post

Description generated with very high confidence]()

**Read a file which contains one word in each line. Then Calculate the character frequency in each line**

![A screenshot of a social media post

Description generated with very high confidence]()

![A screenshot of a social media post

Description generated with very high confidence]()

**Draw a Game board**

**--- --- ---**

**| | | |**

**--- --- ---**

**| | | |**

**--- --- ---**

**| | | |**

**--- --- ---**

**This one is 3x3 (like in tic tac toe). Obviously, they come in many other sizes (8x8 for chess)**

**Ask the user what size game board they want to draw, and draw it for them to the screen using**

**Python’s print statement. Hint: use functions.**

**heightinp= int(input("Enter the height of the board: "))**

**widthinp= int(input("Enter the width of the board: "))**

**board\_draw(heightinp,widthinp)**

![A screenshot of a social media post

Description generated with very high confidence]()