Reports summary:

- The highest CPU usage I could get is around 50%-60%, which is classed as “normal”
- The highest memory I could get is around 98%- classed as “busy”
    - Most 1hr reports could not be found, so I had to run some 1min long reports.

- Min memory I got was around 50% - but this depends on the machine Average was about 70% or so.
- CPU - low - 16% and high - 50%. Average around 35%

- I tried running everything on the computer, and trying different things, but I could only get the CPU usage to be about 50% max. The memory was heavily impacted though. Not sure where to go from here exactly.

Commit:
This represents the amount of virtual memory that has been reserved for a process and has been committed (i.e., allocated). It includes both physical memory (RAM) and disk-based virtual memory (paging file). Commit KB indicates the total memory that the process is guaranteed to access.

Working Set:
This refers to the amount of physical memory currently being used by a process. It includes the pages that are currently in RAM and can be accessed without any additional paging. The working set can change frequently as the process runs and as memory is allocated or freed.

Shareable:
This indicates the portion of the working set that can be shared among multiple processes. For example, shared libraries (DLLs) can be loaded into memory once and shared across processes, reducing memory usage.

Private:
This refers to the portion of the working set that is unique to a specific process and cannot be shared with other processes. Private memory is used for data that the process owns exclusively.

**Working set is the one I will parse**
**Also parse CPU usage and see that fluctuating**

**Do for this upcoming week:**
Upload weekly reports to github
Parse that data (find a way, like python)
Find it for now, do all processes and parse them output as like a .txt or something
Want to make a script that will compare the average data for the samples, and compare it to a new one, and set parameters for mins and maxes, ex. 40% beyond what is is now
So if a process exceeds 40% vs. its average, it will trigger a notification or something to let the user of the machine know
Parse like 10 reports or whatever, find an average, and set up ANOTHER script that parses the data, but for new reports, and compare the numbers with the previous reports, and if its like 40% greater or something, it will be noted

(weekly task it to basically just parse data, make comparison script, old vs new data)