Python File Handling & Utilities – Practice Set

This set is based on the topics we've covered so far:

- **File I/O in Python**
- **Read, Write, and Append Files**
- **OS and Shutil Modules in Python**
- **Creating Command Line Utilities**

These exercises will help you practice working with files, directories, and building simple command-line tools.

1. File I/O Basics

1. Create a text file `notes.txt` using Python and write `"Learning Python is fun!"` into it.
2. Open `notes.txt`, read its content, and print it to the console.

2. Read, Write, and Append Files

1. Write a program that writes three lines of text to a file `tasks.txt`.
2. Open `tasks.txt` in **append mode** and add a new line `"Task Completed!"`.
3. Read the file and print all lines as a list using `readlines()`.

3. OS and Shutil Modules

1. Use the `os` module to:

    1. Print the current working directory
    2. List all files and folders in the current directory
    3. Create a new folder `my_folder`

2. Use the `shutil` module to:

    1. Copy a file from one folder to another
    2. Move a file to a new folder
    3. Delete a file (careful: irreversible!)

---

4. Creating Command Line Utilities

1. Write a small script `count_lines.py` that takes a filename as input and prints how many lines are in the file.
   Example usage:

   ```
   python count_lines.py tasks.txt
   # Output: Number of lines: 4
   ```

2. Write a command-line utility `search_word.py` that takes two arguments:

    1. A filename
    2. A word to search and prints how many times the word appears in the file.

---

5. Bonus Challenges

1. Write a program that reads a file and creates another file with **all words converted to uppercase**.

2. Create a script that deletes all `.tmp` files from the current directory using `os` and `os.remove()`.

3. Write a Python command-line tool that takes a folder name and prints the total size of all files inside it (use `os.path.getsize()`).