Wexdi SoftSkills

File Handling

What is file Handle?

- File handling in Python is a critical skill for developers and involves several operations:
  - **Reading**:
    - Using 'r' mode to read from an existing file.
  - **Writing**:
    - Employing 'w' mode to write to a file, which may create a new file or overwrite an existing one.
  - **Appending**:
    - Utilizing 'a' mode to add to a file without changing what's already there.
  - **Reading and Writing**:
    - Applying 'r+', 'w+', or 'a+' modes for both reading and writing activities

Opening a File

- You can open a file using the **open()** function. It returns a file object, which you can use to read, write, or manipulate the file.

```
file = open("example.txt", "r")  # Opens the file in read mode
```

Reading from a File

- **Read the entire file:**

```
content = file.read()
print(content)
```

- **Read line by line:**

```
for line in file:
    print(line.strip())  # strip() removes any trailing newline characters
```

- **Read specific number of characters:**

```
content = file.read(10)  # Reads the first 10 characters
print(content)
```

Writing to a File

- **Write to a file:**

```
file = open("example.txt", "w")
file.write("Hello, World!")
file.close()
```

- **Append to a file:**

```
file = open("example.txt", "a")
file.write("\nAppending this line.")
file.close()
```

Managing Files

- You can also manage files using various methods:

  - **Check if a file is readable or writable:**

    ```
    file.readable()  # Returns True if the file is readable
    file.writable()  # Returns True if the file is writable
    ```

  - **Get the current file position:**

    ```
    position = file.tell()
    print(position)
    ```

  - **Move the file pointer to a specific position:**

    ```
    file.seek(0)  # Moves the file pointer to the beginning of the file
    ```

Using the with Statement

- It’s a good practice to use the with statement when working with files.
- It ensures that the file is properly closed after its suite finishes, even if an exception is raised.

```
with open("example.txt", "r") as file:
    content = file.read()
    print(content)
```