

## With statement in file handling

File handling in programming involves a set of functions or statements to open, read, write, and close files. Here's an illustration with Python statements commonly used in file handling:

1.

### Opening a file:

The `open()` function opens a file for reading, writing, or appending.

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

### Reading from a file:

Various methods exist to read from a file, such as `read()`, `readline()`, or `readlines()`.

```
content = file.read() # Reads the entire file content
```

### Writing to a file:

Using `write()` allows adding text to a file opened in write ("w") or append ("a") mode.

```
file = open("example.txt", "w")  
file.write("Hello, File Handling!") # Writes to the file
```

### Closing a file:

The `close()` method ensures all file operations are complete and the file is properly saved.

```
file.close()
```

## Using with statement:

The `with` statement is a context manager that automatically handles file closure.

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

**Appending to a file:**

The append mode "a" allows adding content to the end of an existing file without overwriting it.

```
with open("example.txt", "a") as file:  
    file.write("\nAdditional line")
```

These basic statements form the foundation of file handling and can be expanded for more complex operations.