



File Handling in Python - Interview Preparation

Objective:

Prepare for Python interviews with an interactive approach to **file handling** by covering key concepts, practical examples, and frequently asked interview questions.

1. What is File Handling in Python?

File handling in Python allows you to perform operations such as:

- Create
 - Read
 - Write
 - Append
 - Delete files
-



2. File Handling Modes

Python provides different modes to open a file.

Mode	Description	Example Usage
<code>r</code>	Read (default mode)	<code>file.read()</code>
<code>w</code>	Write (creates a file if not exist)	<code>file.write()</code>
<code>a</code>	Append (adds content to the end)	<code>file.write()</code>

x	Create (fails if the file exists)	<code>file.write()</code>
rb	Read in binary mode	<code>file.read()</code>
wb	Write in binary mode	<code>file.write()</code>

3. File Opening and Closing

➤ Opening a File

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

➤ Closing a File

```
file.close()
```

4. Reading from a File

➤ 1. Using `read()` Method

```
file = open("example.txt", "r")
content = file.read()
print(content)
file.close()
```

➤ 2. Using `readline()` Method

```
file = open("example.txt", "r")
line1 = file.readline()
print(line1)
file.close()
```

➤ 3. Using `readlines()` Method

```
file = open("example.txt", "r")
```

```
lines = file.readlines()
for line in lines:
    print(line, end="")
file.close()
```

5. Writing to a File

➤ 1. Using `write()` Method

```
file = open("example.txt", "w")
file.write("Hello, Python!\n")
file.write("File handling is easy.")
file.close()
```

➤ 2. Using `writelines()` Method

```
file = open("example.txt", "w")
lines = ["Python is fun!\n", "Let's learn file handling.\n"]
file.writelines(lines)
file.close()
```

6. Appending to a File

```
file = open("example.txt", "a")
file.write("\nThis content is appended.")
file.close()
```

7. File Creation and Deletion

➤ 1. Creating a File

```
file = open("newfile.txt", "x")
file.close()
```

➤ 2. Deleting a File

```
import os
if os.path.exists("newfile.txt"):
    os.remove("newfile.txt")
    print("File deleted successfully.")
else:
    print("File does not exist.")
```

8. Common Interview Questions

1. What is the difference between **r+**, **w+**, and **a+** modes?

- **r+** - Read and write; raises an error if the file does not exist.
 - **w+** - Write and read; creates a new file if it does not exist and overwrites content.
 - **a+** - Append and read; creates a new file if it does not exist and appends content.
-

2. How do you read a file without using **close()**?

Using **with** statement:

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

 *Auto-closes the file after the block execution.*

3. How to check if a file exists in Python?

```
import os
print(os.path.exists("example.txt")) # True if file exists
```

9. Advanced Concepts in File Handling

➤ 1. Working with Binary Files

```
with open("image.jpg", "rb") as file:
    data = file.read()
    print(data[:20]) # Read first 20 bytes
```

➤ 2. Copying File Content

```
with open("source.txt", "r") as src:
    content = src.read()

with open("destination.txt", "w") as dest:
    dest.write(content)
```

➤ 3. Error Handling in File Operations

```
python
CopyEdit
try:
    file = open("nonexistent.txt", "r")
except FileNotFoundError:
    print("File not found!")
```

10. Practice Challenges

1. Count the Number of Lines in a File

```
with open("example.txt", "r") as file:
    lines = file.readlines()
```

```
print("Total lines:", len(lines))
```





2. Copy Content of One File to Another

```
with open("source.txt", "r") as src:  
    content = src.read()  
  
with open("destination.txt", "w") as dest:  
    dest.write(content)
```

3. Search for a Word in a File

```
word_to_find = "Python"  
with open("example.txt", "r") as file:  
    content = file.read()  
    if word_to_find in content:  
        print(f"'{word_to_find}' found in file.")  
    else:  
        print(f"'{word_to_find}' not found in file.")
```

11. Quick Tips for Interview Success

-  Understand different file modes (`r`, `w`, `a`, `x`, `rb`, `wb`).
 -  Always use `with` for safe file handling.
 -  Know how to handle exceptions using `try-except` in file operations.
 -  Be familiar with `os` module for file management.
-