TECHWINGSYS™

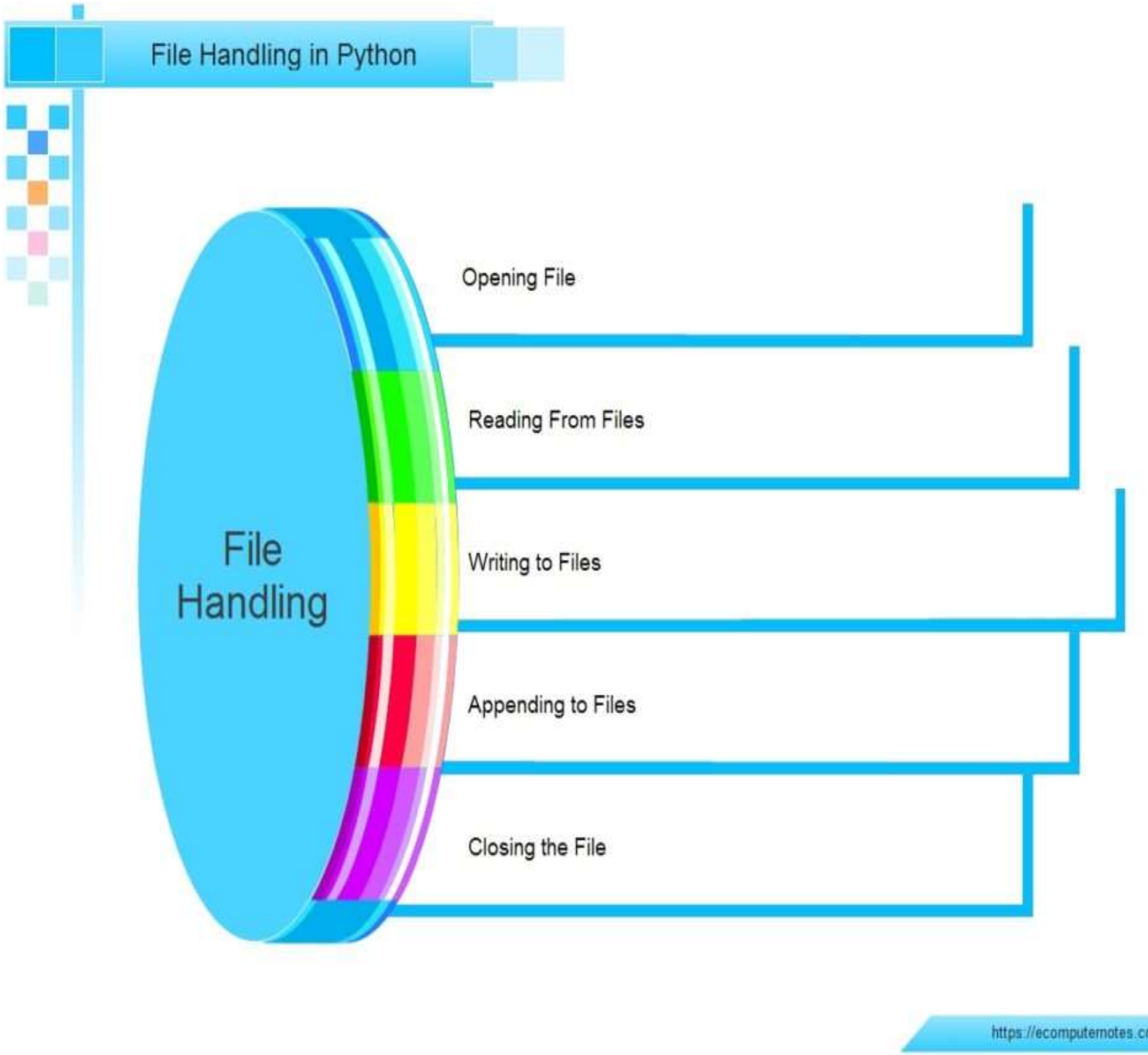
5th Floor, Usnaz Tower, Church Landing Rd, near Medical Trust Hospital, Pallimukku, Kochi, Kerala 682016

File I/O Handling

File Handling in Python

File handling is an important activity in every web app. The types of activities that you can perform on the opened file are controlled by Access Modes. These describe how the file will be used after it has been opened.

These modes also specify where the file handle should be located within the file. Similar to a pointer, a file handle indicates where data should be read or put into the file.



https://ecomputernotes.c

1.Read ('r’): This mode opens the text files for reading only. The start of the file is where the handle is located. It raises the I/O error if the file does not exist. This is the default mode for opening files as well.

2.Write ('w’): This mode opens the file for writing only. The data in existing files are modified and overwritten. The start of the file is where the handle is located. If the file does not already exist in the folder, a new one gets created.

3.Append ('a’): This mode allows the file to be opened for writing. If the file doesn't yet exist, a new one gets created. The handle is set at the end of the file. The newly written data will be added at the end, following the previously written data.

Opening a file for reading

You can open a file using the open() function in Python. The open() function takes two arguments, the file name and the mode in which you want to open the file. The mode can be "r" for reading, "w" for writing, "a" for appending, or "x" for exclusive creation.

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

Reading the contents of the file

You can read the contents of a file using the read() method. You can read the entire contents of a file or a specific number of characters.

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

Printing the contents of the file

```
print(content)
```

Opening a file for writing

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

Writing content to the file

You can write to a file using the write() method. This method allows you to write a string to the file.

```
file.write("This is a sample text.")
```

Opening a file for appending

You can append to a file using the append() method. This method allows you to add new content to the end of the file.

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

Appending content to the file

```
file.write("This is some additional text that we are appending to the file.")
```

Closing the file

After performing operations on a file, it is important to close the file using the close() method to free up resources.

```
file.close()
```