RED & WHITE®

Multimedia Education

Shaping "skills" for "scaling" higher...!!!

WELCOME,
PROGRAMMERS

</>

JAVA

C

C++

01.

WHAT IS
FILE HANDLING?

C LANG.

FILE HANDLING

File handling involves **operations on files**, such as **reading from** or **writing to files**.

The **stdio.h** library provides functions for file handling.

OPERATIONS ON FILES

OPEN A FILE

1. **fopen()** function

READ A FILE

2. **fgets()** or **fscanf()** function

WRITE/APPEND INTO A FILE

3. **fputs()** or **fprintf()** functions

CLOSE A FILE

4. **fclose()** function

RED & WHITE®
Multimedia Education
Shaping "skills" for "scaling" higher...!!!

FILE OPENING MODES

"r"

File Reading

"a"

**File Writing
(Appending)**

"w"

File Writing

"x"

File Creation

Let's see each **operations** in detail...

01

OPEN A FILE

Open a File

To open a file, you can use the **fopen()** function.
Refer the below mentioned syntax:

```
fopen("fileName", "mode");
```

filename is the **name of the file**, and **mode** specifies the **type of access**
(e.g., "r" for reading, "w" for writing, "a" for appending).

Open a File

The **fopen()** function returns an address of a type **FILE**.
So to receive this, we have to create a ***FILE** pointer.

```
FILE *ptr = fopen("fileName", "mode");
```

02

READ A FILE

Read a File

To read a file, you can use the **fgets()** function.
Refer the below mentioned syntax:

```
fgets(container, size of buffer, filePointer);
```

size of buffer is the **size of characters** that reads/fetches from a file at a time.

If all characters are already fetched,
then **fgets()** function returns **NULL**.

```
fgets(container, size of buffer, filePointer);
```

03

WRITE INTO A FILE
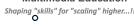
Write into a File

To write into a file, you can use the **fputs()** function. Refer the below mentioned syntax:

```
fputs("data", filePointer);
```

Note: In write mode, every time the data written into a file, old content are getting wiped out.

Append into a File

To append into a file, you can use the **fputs()** function. Refer the below mentioned syntax:

```
fputs("data", filePointer);
```

Note: In append mode, every time the data written into a file, old content are getting stored as it is.
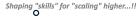
Close a File

To close a connection from a file, you can use the **fclose()** function.

Refer the below mentioned syntax:

```
fclose(filePointer);
```

LANGUAGE

Let's start now...