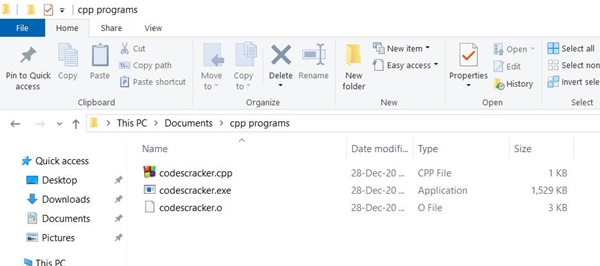
**Program to Write Date into a File**

In this, you will learn and get code to write some data entered by user into a file using a C++ program. In this article we've created two programs on the same topic. The first program is the simplest one to write data into a file. Whereas the second program is the modified version of first program.

To write some content in a file using C++ programming, you have asked from user to enter the name of file along with its extension say **codescracker.txt** or **codescracker.html** etc.

Now we must have to open the file in writing mode. Therefore, if file doesn't exist, then the file automatically gets created inside the current directory.

The current directory, means the directory where you are saving your C++ source code. Because I'm saving my C++ source code in **cpp programs** folder. Here is the snapshot of the folder, before executing the program given below:



**Write Content in a File**

Now let's go through the program given below that shows how a file gets created and data entered by user at run-time gets written into the file. The program is created in a way that, user can write as much data as they wants, into the file:

// C++ Program to Write Data into a File

// ----codescracker.com----

#include<iostream>

#include<stdio.h>

#include<fstream>

#include<string.h>

using namespace std;

int main()

{

char fname[20], str[200];

fstream fp;

cout<<"Enter the Name of File: ";

gets(fname);

fp.open(fname, fstream::out);

if(!fp)

{

cout<<"\nError Occurred!";

return 0;

}

cout<<"Enter the Data: ";

gets(str);

while(strlen(str)>0)

{

fp<<str;

fp<<"\n";

gets(str);

}

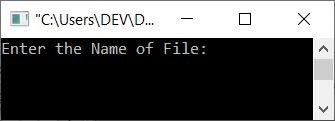
fp.close();

cout<<endl;

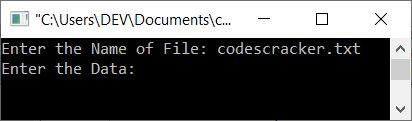
return 0;

}

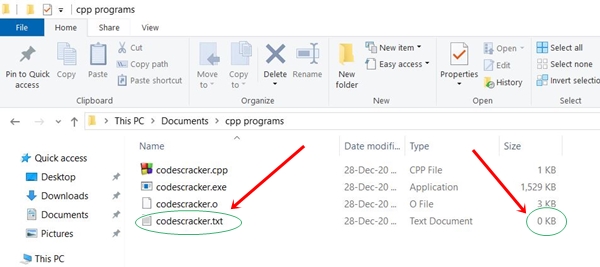
This program was build and run under *Code::Blocks* IDE. Here is the initial output produced by this C++ program:



Now enter the name of file say **codescracker.txt** and press ENTER key as shown in the snapshot given below:



Here is the snapshot of the folder **cpp programs**:



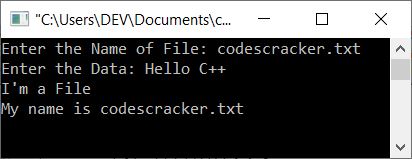
As you can see, the file *codescracker.txt* gets created. It is newly created file with empty content. Therefore, its size is shown as **0KB**. Let's put some content in it through the program given below. Now supply the following data:

Hello C++

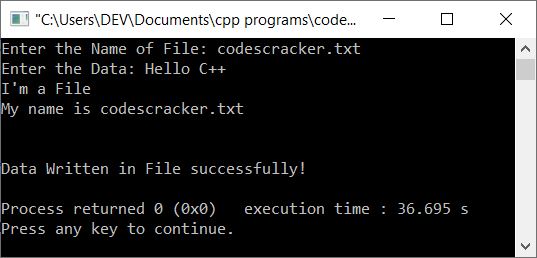
I'm a File

My name is codescracker.txt

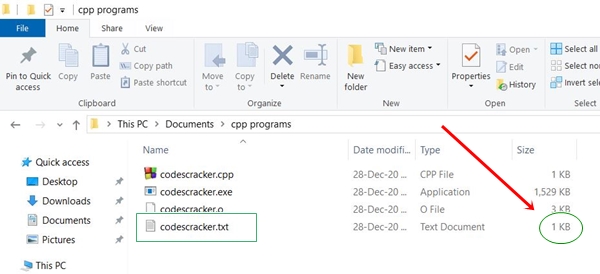
as shown in the snapshot given below:



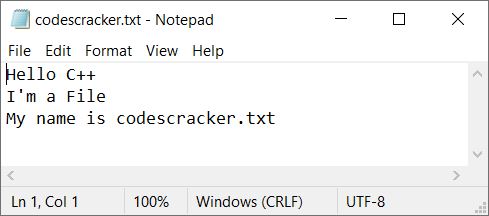
After writing these three lines of data, press two times ENTER key to put these data into the file *codescracker.txt*. Here is the output shown after pressing *ENTER* key:



Here is the snapshot of current directory:



As you can see that the size of the file gets increased. And here is the snapshot of opened file *codescracker.txt* file:



**Note -**The **open()** function is used to open the file. It takes two arguments, the filename and its opening mode. The first argument (filename) is compulsory. We've opened the file in writing mode using **fstream::out**.

**Note -**Don't forgot to [close the file](https://codescracker.com/cpp/cpp-opening-closing-files.htm) using **close()** function.

The condition (of **while loop**) **strlen(str)>0** checks whether the length of entered string by user is greater than 0 or not. If it greater than 0, means user has entered some data. Therefore condition evaluates to be true and content or data gets written inside the file, using **fp<<str** statement.

And a newline also gets written to the file. Then again we've received the string and checks whether its length is greater than 0 or not. In this way, data are getting received by user and gets written into the file continuously until user pressed ENTER key without entering anything.

**Modified Version of Previous Program**

This is the modified version of previous program. This program prints message for each step going inside the program:

// C++ Write Data into a File Modified Version

// ----codescracker.com----

#include<iostream>

#include<stdio.h>

#include<fstream>

#include<string.h>

using namespace std;

int main()

{

char fname[20], str[200];

fstream fp;

cout<<"Enter the Name of File: ";

gets(fname);

fp.open(fname, fstream::in);

if(!fp)

{

cout<<"\nFile Doesn't Exist!\n";

cout<<"\nCreating the File...\n";

fp.open(fname, fstream::out);

if(!fp)

{

cout<<"\nError Occurred while Creating the File!";

cout<<"\nExiting...";

return 0;

}

else

cout<<"\nFile Created Successfully!";

}

fp.close();

fp.open(fname, fstream::out);

if(!fp)

{

cout<<"\nError Occurred while Opening the File!";

cout<<"\nExiting...";

return 0;

}

cout<<"\nEnter the Data: ";

gets(str);

while(strlen(str)>0)

{

fp<<str;

fp<<"\n";

gets(str);

}

cout<<"\nData Written in File successfully!";

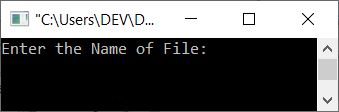
fp.close();

cout<<endl;

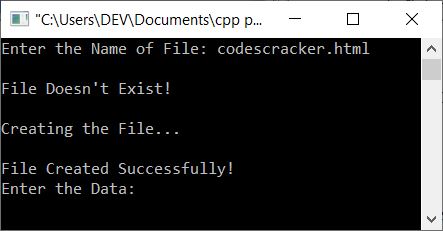
return 0;

}

Here is the initial output produced by this C++ program:



Now enter the name of file say *codescracker.html* and press *ENTER* key. Here is the sample output after doing this:



Now enter the following data:

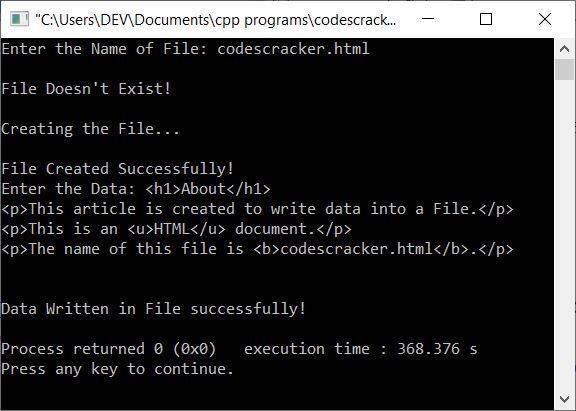
<h1>About</h1>

<p>This article is created to write data into a File.</p>

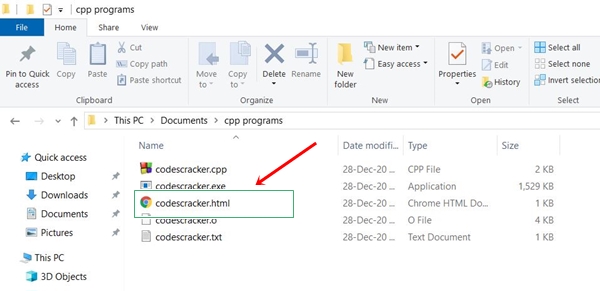
<p>This is an <u>HTML</u> document.</p>

<p>The name of this file is <b>codescracker.html</b>.</p>

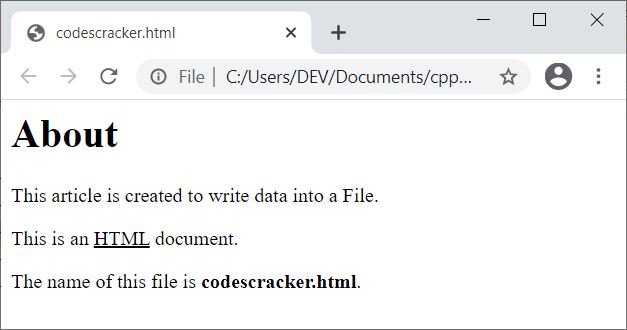
After supplying these inputs one by one, press double times *ENTER* key, here is the output you will see:



If you see the folder, **cpp programs**, then the file *codescracker.html* will be available there as shown in the snapshot given below:



And if you open this file into a browser say in **Google Chrome**, then it looks like:



But if you open the same file in a text editor like **Notepad**, then it looks like:

