

# Reading from a File

May 2, 2020

## 0.1 File Input Streams

### 0.1.1 Creating an Input Stream Object

In C++, you can use the `std::ifstream` object to handle input file streams. To do this, you will need to include the header file that provides the file streaming classes: `<fstream>`.

Once the `<fstream>` header is included, a new input stream object can be declared and initialized using a file path `path`:

```
std::ifstream my_file;  
my_file.open(path);
```

Alternatively, the declaration and initialization can be done in a single line as follows:

```
std::ifstream my_file(path);
```

C++ `ifstream` objects can also be used as a boolean to check if the stream has been created successfully. If the stream were to initialize successfully, then the `ifstream` object would evaluate to `true`. If there were to be an error opening the file or some other error creating the stream, then the `ifstream` object would evaluate to `false`.

The following cell creates an input stream from the file "files/1.board":

```
In [ ]: #include <fstream>  
        #include <iostream>  
        #include <string>  
  
        int main()  
        {  
            std::ifstream my_file;  
            my_file.open("files/1.board");  
            if (my_file) {  
                std::cout << "The file stream has been created!" << "\n";  
            }  
        }
```

Compile & Execute

Explain

Loading terminal (id\_7qs09au), please wait...

### 0.1.2 Reading Data from the Stream

If the input file stream object has been successfully created, the lines of the input stream can be read using the `getline` method. In the cell below, a while loop has been added to the previous example to get each line from the stream and print it to the console.

```
In [ ]: #include <fstream>
        #include <iostream>
        #include <string>

        int main() {
            std::ifstream my_file;
            my_file.open("files/1.board");
            if (my_file) {
                std::cout << "The file stream has been created!" << "\n";
                std::string line;
                while (getline(my_file, line)) {
                    std::cout << line << "\n";
                }
            }
        }
```

Compile & Execute

Explain

Loading terminal (id\_6gag5bi), please wait...

### 0.1.3 Recap

That's it! To recap, there are essentially four steps to reading a file: 1. `#include <fstream>` 2. Create a `std::ifstream` object using the path to your file. 3. Evaluate the `std::ifstream` object as a `bool` to ensure that the stream creation did not fail. 4. Use a while loop with `getline` to write file lines to a string.

## 0.2 On to an Exercise

Have a careful look at the code above for reading file lines into the code. In the next exercise, you will write a function to do this in your program.