Solution: Write the Contents of a File to Another File

This lesson provides a solution to the challenge given in the previous lesson.

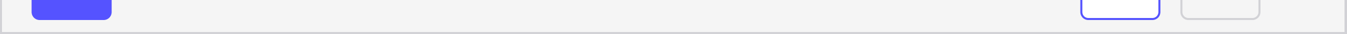
WE'LL COVER THE FOLLOWING

- Solution
- Solution explanation

Solution

Here is the code that will write the contents of a file to another file.

```
import std.stdio;
import std.string;

void main() {

    string inFileName = "foo.txt";
    File inFile = File(inFileName, "r");

    string outFileName = inFileName ~ ".out";
    File outFile = File(outFileName, "w");

    while (!inFile.eof()) {
        string line = strip(inFile.readln());

        if (line.length != 0) {
            outFile.writeln(line);
        }
    }

    writeln(outFileName, " has been created.");

    inFile = File(outFileName, "r");

    while (!inFile.eof()) {
        string line = strip(inFile.readln());
        writeln(line);
    }

}
```



Code to write contents of a file to another file

Solution explanation

- **Line 7:**

  ```
  File inFile = File(inFileName, "r");
  ```

  The name of the file is stored in the `inFileName` variable, so we open the file with this name in read mode by specifying `r` in the parameters of the `File` function.

- **Line 9:**

  ```
  string outFileName = inFileName ~ ".out";
  ```

  As per the requirement of the challenge, the output file name should be the name of the input file name with `.out` appended at the end. Hence, we are using `~` operator here to append `.out` at the end of the input file name.

- **Line 10:**

  ```
  File outFile = File(outFileName, "w");
  ```

  Since we only intend to write data in the output file, we open it in the write mode by specifying the `w` character in the parameters.

- **Line 12 and 13:**

  ```
  while (!inFile.eof()) {
      string line = strip(inFile.readln());
  ```

  Here we are using the `.eof()` and `strip()` functions to keep reading the contents of the file until the end of the file is reached.

In the next lesson, you will find a quiz related to file handling concepts covered in this lesson.