Logo

Description automatically generated

**22/SU-COP-2805C-72035 Java Programming**

**Exercise 12.21**

Document Version: 0.1

Version Date: July 9, 2022

Created By: Johnathan Webster

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Rationale |
| 0.1 | 2022 July 9 | Johnathan Webster | First Draft |

# Document Purpose

The purpose of this document is to define the updated DataSorted() class.

# Technical Specifications

## Purpose of Technical Implementation

The purpose of this implementation is to open a text file and read if all information in the text file is sorted.

## Technical Implementation Components

The class implementation allows the designer to validate if the information in a text file is sorted in alphabetical order.

First we check to see if the file already exists and exit if it does:

// *Check if file exists*

*if* (!file.*exists*()) {

            System.*out*.*println*(file.*getName*() + " does not exist");

            System.*exit*(0);

        }

Then if the file does exist to read the information in the file:

*try* (

                // *Create input file*

                Scanner input = *new* *Scanner*(file);) {

            // *Read first string from file*

*if* (input.*hasNext*())

                string1 = input.*next*();

Then, if the string we have assigned to the first input is to keep incrementing:

*while* (input.*hasNext*() && dataSorted) {

                string2 = input.*next*();

If they are not in order, then return error message:

*if* (string1.*compareTo*(string2) > 0) {

                    System.*out*.*println*(

                            "The strings " + string1 + " and " + string2 +

                                    " are out of order");

                    dataSorted = false;

                }

                string1 = string2;

## DataSorted Class Implementation

The implementation of this class allows the designer to validate if the specified file does exist and if it does, then to provide if the data is sorted in the file.