Objectives:

* Writing and reading files
* Try and catch statement
* Intro to GUI Interfaces

**There are 8 challenge exercises with 8 print screens, each worth 12.5%**

Please submit this document for grading when completed… Please work in-groups.

**Project #1** (Writing and Reading files)

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated

Text

Description automatically generatedA picture containing logo

Description automatically generated

**Challenge Exercise #1:** write a program that will ask the user to enter the first, last name with the age, and write the contents to a file, and then read from the file.

**#1 print screen the running application with code below here.**

**Project #2** (Writing numeric data)

Text

Description automatically generated with medium confidence

**Challenge Exercise #2:** continuing with project #2, read the data using the print statement.

**#2 print screen the running application with code below here.**

**Project #3** (Writing and reading the Sales Data)

Text, letter

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

**Challenge Exercise #3:** continuing from project #3, write a program that will ask the user to enter the sales with salary, and if the total sales are greater than 1000 add 10% commission to the salary write and read the data using the print statement.

*Please enter the following test data: enter 80,000 for the salary, enter the days of sale as 5, each sale entered will be 1000, see below*

Text

Description automatically generated

**#3 print screen the running application with code below here.**

**Project #4** (Write Records)

Text

Description automatically generated

**Challenge Exercise #4:** Continuing with project #4, read the content of the employees.txt file

**#4 print screen the running application with code below here.**

**Project #5:** using the try and catch statement with the while loop

Graphical user interface, text, application, email

Description automatically generated

**Project #5 (**Introducing GUI interfaces using Tkinter)

**Text

Description automatically generatedText

Description automatically generated**

**Challenge Exercise #5:** Continuing with project #5, add the address, city, state with zip code and transfer the information to a text file.

**#5 print screen the running application with code below here.**

**Challenge Exercise #6:** create a **GUI** interface that will write 3 numbers and sum + averages the total into a text file.

**Example output in the text file**

The three numbers are: 1, 2 and 3

The total is 6

The average is 2

**#6 print screen the running application with code below here.**

**Challenge Exercise #7:** create a console (**NO GUI**) application that will ask the user to enter the full name, with the grade of the midterm and final exam and **write** the **average** grade with **letter** grade into a text file. Use the table as a guide. Use also the try statement for this application. Make sure to append the data and **read** the content of the file also.

Use the

|  |  |
| --- | --- |
| 90-100 | Letter grade A |
| 80-89 | Letter grade B |
| 70-79 | Letter grade C |
| 60-69 | Letter grade D |
| Under 60 | Letter grade F |

**#7 print screen the running application with code below here.**

**Challenge Exercise #8:** create a console application (**NO GUI**) that will write random numbers from 1-10 into a text file and read the contents.

**#8 print screen the running application with code below here.**

**Submit this document to Module 6 Class Exercise.**