



Republic of the Philippines
Laguna State Polytechnic University
Province of Laguna



College of Computer Studies

**KnowYourState: A Survey about CCS Freshman
Students' Mental Well-being Using C# Console
Application.**

Submitted by:

Jamolin, Leigh Inna C.
Venancio, Nicko A.
Casiño, Kurt Sebastian F.
BSCS 1-A

Submitted to:

De Jesus, Mervin Jommel T.
Instructor of ITEC102

I. Introduction

The unexpected situation of the COVID-19 pandemic has caused the closure of universities worldwide and has forced the transition to online learning. This exceptional context compels us to understand students' experience with online learning.

Previous literature identifies relevant factors that intervene in the online education experience and can affect students' academic development. One of the main concerns is the students' mental health, given the lockdown restrictions under which classes have been conducted.

Furthermore, the impact of the prolonged lockdown and the pandemic fatigue on university students and their academic experience is still unclear. There are a number of studies associating student involvement and achievement with mental health and vice versa, and have documented the positive effects of social and emotional learning programming on students of diverse backgrounds. Improved social and emotional behaviors among students can have a strong impact on success in school and ultimately in life. With this situation, the researchers made a survey program and see what will be the result of the survey.

A. Presentation of the Problem

Mental health includes our emotional, psychological, and social well-being. It affects how we think, feel, and act as we cope with life. It also helps determine how we handle stress, relate to others, and make choices.

This study focused on mental health: its effect on freshmen college students. Specifically, it will answer the following problems

1. What is the status of the mental health of the freshmen student in college?
2. Is it possible to create a survey using the console?

B. Rationale and Justification

This study like to know what is happening to freshmen student's mental health, it created a survey application in C# language for freshmen college students to know and see the statistical results of the questionnaire using what the researchers have learned and see if they can make a survey application that will be based on their first semester.

C. Historical or Background of the Problem

According to Poalses and Bezuidenhout, 2018. The negative mental health consequences of online learning among students can include increased anxiety and absenteeism. These can stem from the increased demand for new technological skills, productivity, and information overload.

By answering the simple information about themselves and questionnaires, the total of the respondents who answered the survey will be added to the total of statistics and see the results of all the surveyees.

The survey application can be used for every freshman college student who wants or is willing to answer the questions about their mental health.

II. Project Structure

The hierarchy graph below shows the structure of the project with its classes and the methods inside each class.

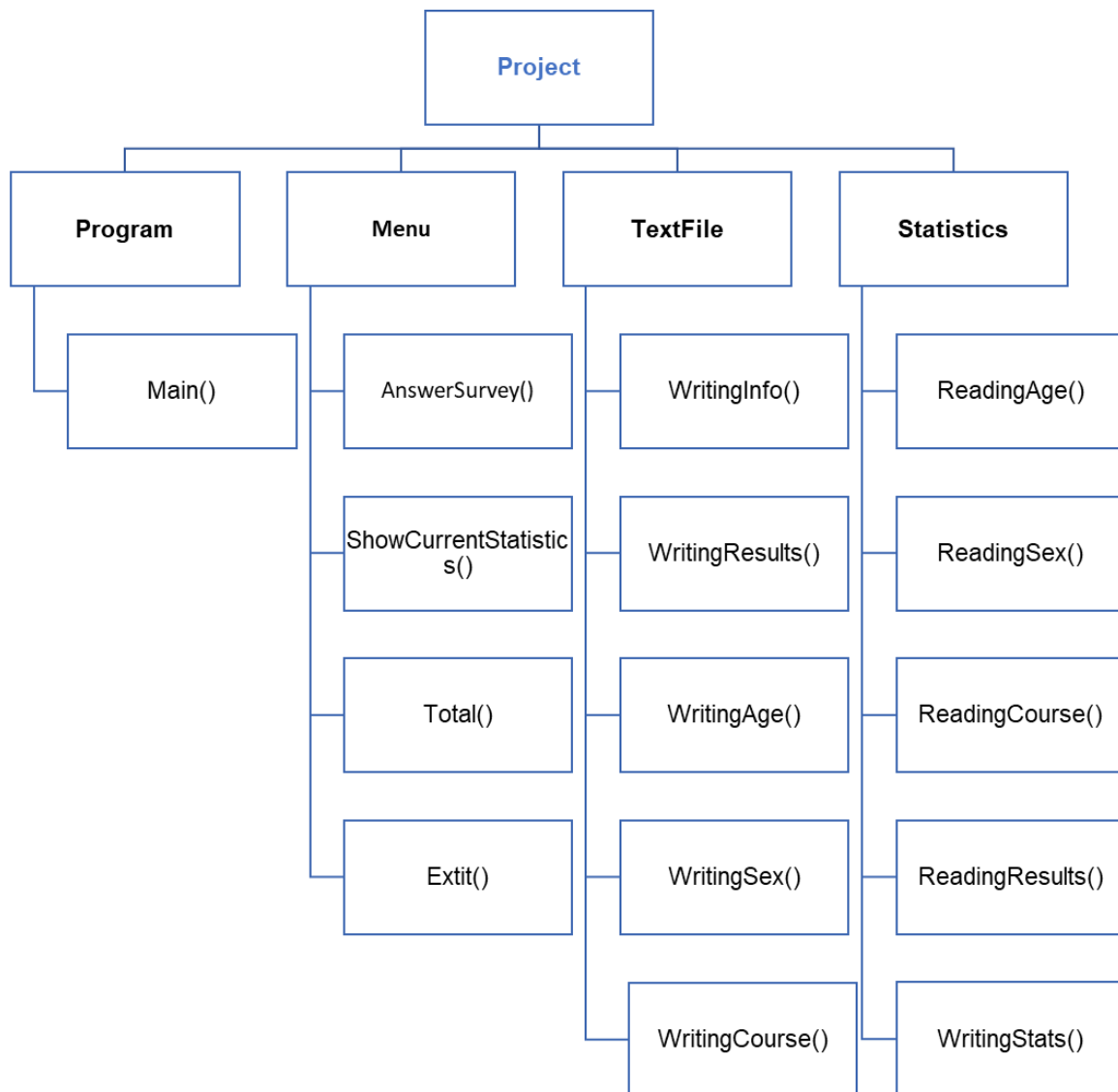


Figure 1. A hierarchical structure of the project

A. Project Modules

This section provides the screenshots of all the modules of the project.

An explanation in every module is stated here such as the function and roles of that module in the project.

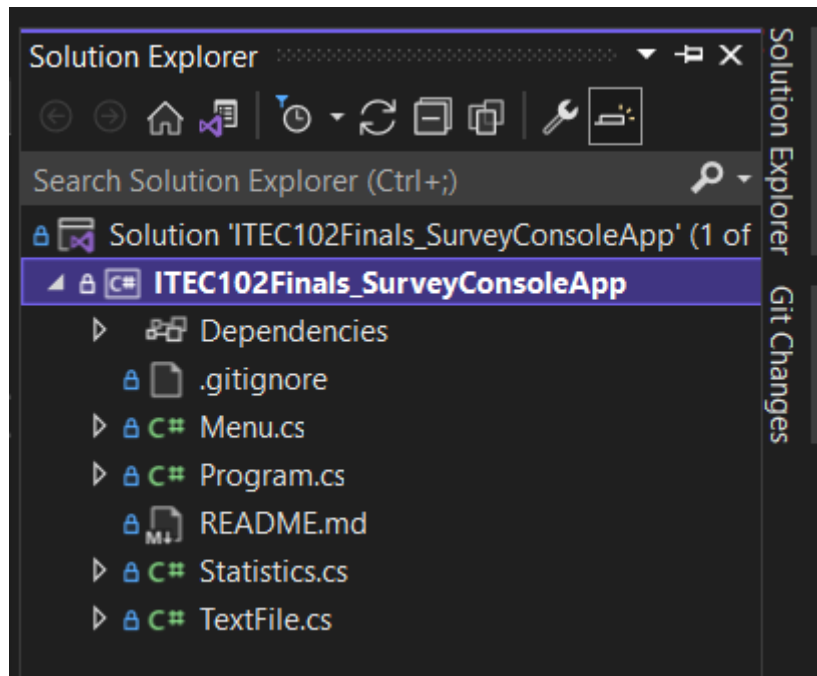
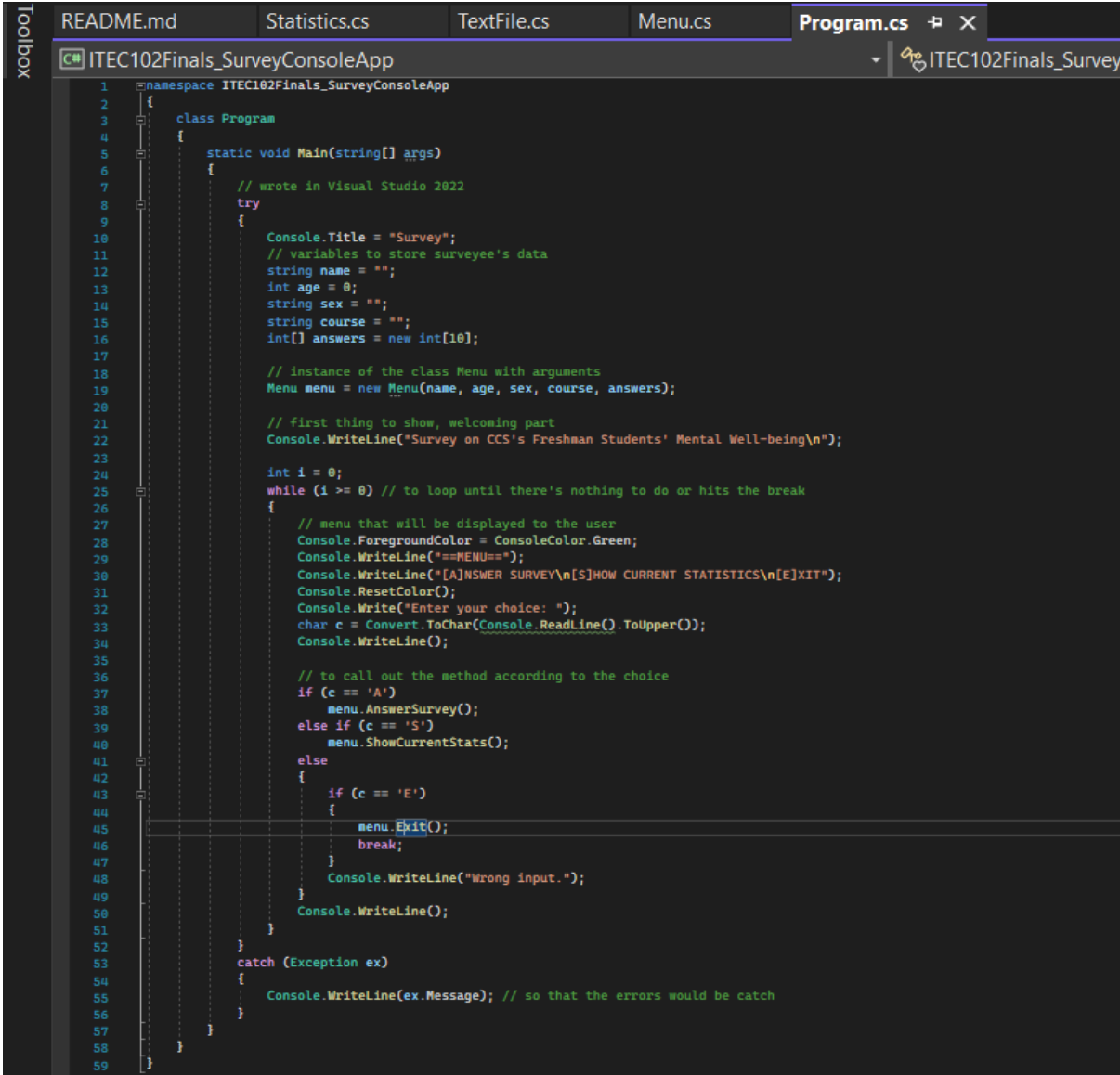


Figure 2. *What inside the program*

The solution file includes .gitignore and README.md for GitHub purposes, the classes are in all separate files: Program, Menu, TextFile, and Statistics.



```
1 namespace ITEC102Finals_SurveyConsoleApp
2 {
3     class Program
4     {
5         static void Main(string[] args)
6         {
7             // wrote in Visual Studio 2022
8             try
9             {
10                Console.Title = "Survey";
11                // variables to store surveyee's data
12                string name = "";
13                int age = 0;
14                string sex = "";
15                string course = "";
16                int[] answers = new int[10];
17
18                // instance of the class Menu with arguments
19                Menu menu = new Menu(name, age, sex, course, answers);
20
21                // first thing to show, welcoming part
22                Console.WriteLine("Survey on CCS's Freshman Students' Mental Well-being\n");
23
24                int i = 0;
25                while (i >= 0) // to loop until there's nothing to do or hits the break
26                {
27                    // menu that will be displayed to the user
28                    Console.ForegroundColor = ConsoleColor.Green;
29                    Console.WriteLine("==MENU==");
30                    Console.WriteLine("[A]NSWER SURVEY\n[S]HOW CURRENT STATISTICS\n[E]XIT");
31                    Console.ResetColor();
32                    Console.Write("Enter your choice: ");
33                    char c = Convert.ToChar(Console.ReadLine().ToUpper());
34                    Console.WriteLine();
35
36                    // to call out the method according to the choice
37                    if (c == 'A')
38                        menu.AnswerSurvey();
39                    else if (c == 'S')
40                        menu.ShowCurrentStats();
41                    else
42                    {
43                        if (c == 'E')
44                        {
45                            menu.Exit();
46                            break;
47                        }
48                        Console.WriteLine("Wrong input.");
49                    }
50                    Console.WriteLine();
51                }
52            }
53            catch (Exception ex)
54            {
55                Console.WriteLine(ex.Message); // so that the errors would be catch
56            }
57        }
58    }
59 }
```

Figure 3. The Program class

Inside the class program, the variables, title, and also the menu that will be shown on the surveyee are the one written here. Other functions were written in separate files to avoid messy and unorganized code.

```
{  
    // wrote in Visual Studio 2022  
    try  
    {  
        Console.Title = "Survey";  
        // variables to store surveyee's data  
        string name = "";  
        int age = 0;  
        string sex = "";  
        string course = "";  
        int[] answers = new int[10];  
  
        // instance of the class Menu with arguments  
        Menu menu = new Menu(name, age, sex, course, answers);  
    }  
}
```

Figure 4. Variables inside the Program.cs

These variables are included in the Program class so the instantiated Menu class won't be error. This part here also has the Console.Title called "Survey". which is the one that will get displayed in the console title bar.

```
// first thing to show, welcoming part  
Console.WriteLine("Survey on CCS's Freshman Students' Mental Well-being\n");  
  
int i = 0;  
while (i >= 0) // to loop until there's nothing to do or hits the break  
{  
    // menu that will be displayed to the user  
    Console.ForegroundColor = ConsoleColor.Green;  
    Console.WriteLine("==MENU==");  
    Console.WriteLine("[A]NSWER SURVEY\n[S]HOW CURRENT STATISTICS\n[E]XIT");  
    Console.ResetColor();  
    Console.Write("Enter your choice: ");  
    char c = Convert.ToChar(Console.ReadLine().ToUpper());  
    Console.WriteLine();  
}
```

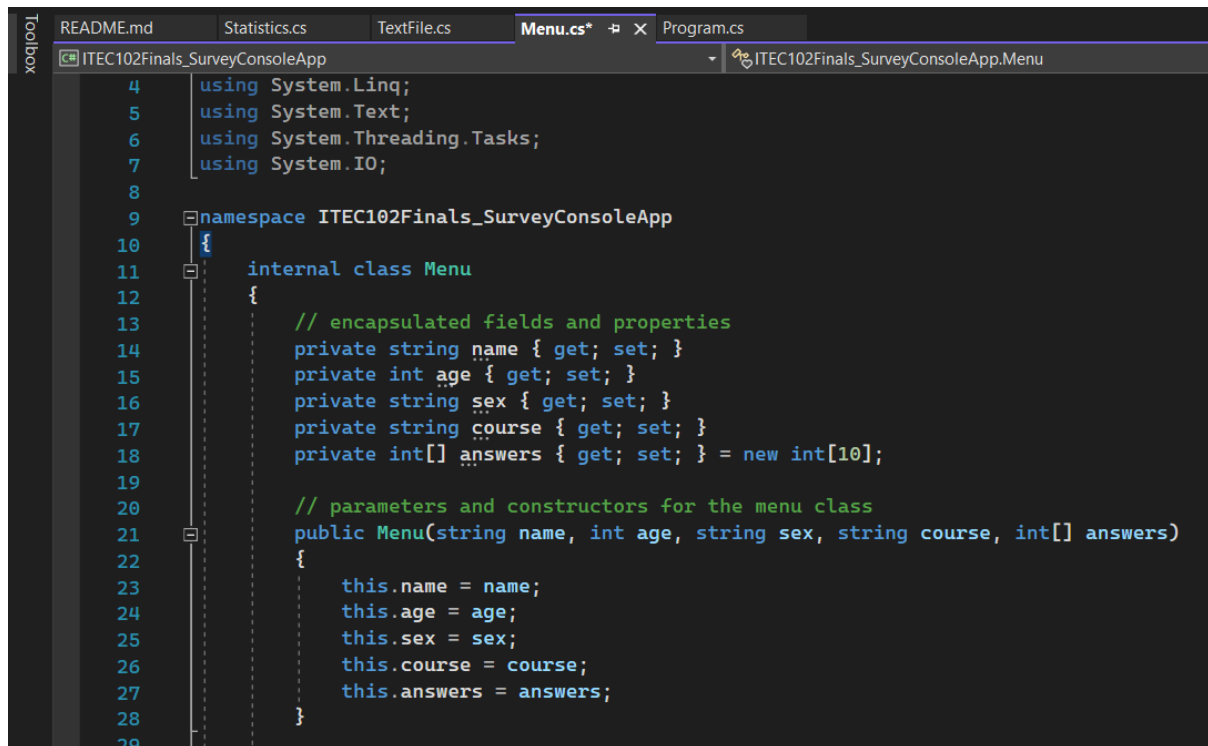
Figure 5. The menu that will be shown

This is where the program will ask the user what to do. The options are answering the survey, showing current statistics, and exiting the program. It's inside a while loop so the user can still select an option after doing one.

```
// to call out the method according to the choice
if (c == 'A')
    menu.AnswerSurvey();
else if (c == 'S')
    menu.ShowCurrentStats();
else
{
    if (c == 'E')
    {
        menu.Exit();
        break;
    }
    Console.WriteLine("Wrong input.");
}
Console.WriteLine();
```

Figure 6. Calling the methods from Menu class to Program class

Using if-else statements to call the functions that the user chose. If they have 'A', it will call the function AnswerSurvey(). 'S' for ShowCurrentStatistics(). If not, it will print out the following text to the console: "Wrong input." However, if it's 'E' it will call the function for exiting the program. The next part of the code is a loop that checks for each letter in turn.



```
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;

namespace ITEC102Finals_SurveyConsoleApp
{
    internal class Menu
    {
        // encapsulated fields and properties
        private string name { get; set; }
        private int age { get; set; }
        private string sex { get; set; }
        private string course { get; set; }
        private int[] answers { get; set; } = new int[10];

        // parameters and constructors for the menu class
        public Menu(string name, int age, string sex, string course, int[] answers)
        {
            this.name = name;
            this.age = age;
            this.sex = sex;
            this.course = course;
            this.answers = answers;
        }
    }
}
```

Figure 7. The Menu class

The Menu class where the option's functions are written. First, the fields are encapsulated because according to (Ganesh, 2019) The need of encapsulation is to protect or prevent the code (data) from accidental corruption due to the silly little errors that we are all prone to make. The constructors with parameters are also present here.

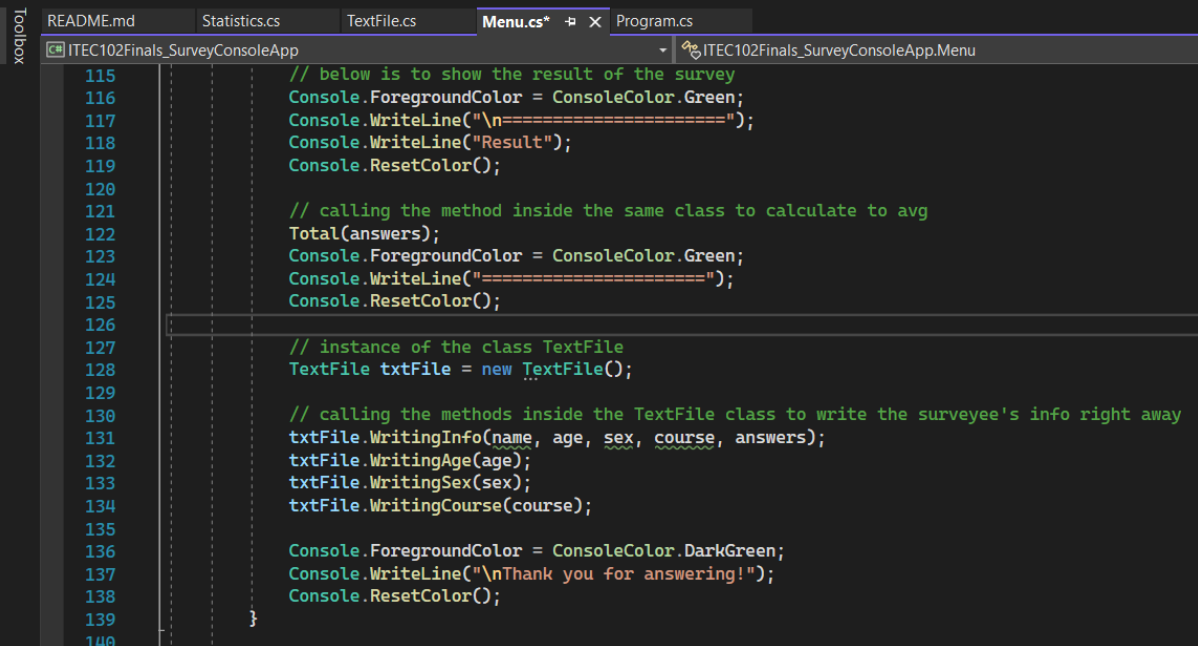
```

28 }
29
30 public void AnswerSurvey()
31 {
32     // getting the information of the surveyee
33     Console.ForegroundColor = ConsoleColor.Green;
34     Console.WriteLine("Student's Information");
35     Console.ResetColor();
36     Console.Write("Name : ");
37     name = Console.ReadLine();
38     Console.Write("Age : ");
39     age = Convert.ToInt32(Console.ReadLine());
40     Console.Write("Sex : ");
41     sex = Console.ReadLine();
42     Console.Write("Course : ");
43     course = Console.ReadLine();
44
45     // instructions for the survey
46     Console.ForegroundColor = ConsoleColor.Green;
47     Console.WriteLine("\nDirections: Please indicate how much you agree or disagree with each of these statements. Just put the number only.");
48     Console.ResetColor();
49     Console.WriteLine("5 - Strongly Agree\n4 - Agree\n3 - Neutral\n2 - Disagree\n1 - Strongly Disagree");
50
51     // these are the questions and where the users will answer
52     Console.ForegroundColor = ConsoleColor.Green;
53     Console.WriteLine("\nStatements:");
54
55     Console.WriteLine("\nDuring this new normal, online class, in the midst of the pandemic, " +
56         "\nI've noticed any changes in my physique, particularly in terms of weight.");
57     Console.ResetColor();
58     Console.Write("Answer: ");
59     answers[0] = Convert.ToInt32(Console.ReadLine());
60
61     Console.ForegroundColor = ConsoleColor.Green;
62     Console.WriteLine("\nI'm losing my appetite.");
63     Console.ResetColor();
64     Console.Write("Answer: ");
65     answers[1] = Convert.ToInt32(Console.ReadLine());
66
67     Console.ForegroundColor = ConsoleColor.Green;
68     Console.WriteLine("\nI have a habit of procrastinating on my schoolwork.");
69     Console.ResetColor();
70     Console.Write("Answer: ");
71     answers[2] = Convert.ToInt32(Console.ReadLine());
72
73     Console.ForegroundColor = ConsoleColor.Green;
74     Console.WriteLine("\nI always stay up late to do my schoolwork.");
75     Console.ResetColor();
76     Console.Write("Answer: ");
77     answers[3] = Convert.ToInt32(Console.ReadLine());
78
79     Console.ForegroundColor = ConsoleColor.Green;
80     Console.WriteLine("\nI was having trouble focusing on my studies and/or work.");
81     Console.ResetColor();
82     Console.Write("Answer: ");
83     answers[4] = Convert.ToInt32(Console.ReadLine());
84
85     Console.ForegroundColor = ConsoleColor.Green;
86     Console.WriteLine("\nMy anxiety grew as a result of the pressure I put on myself to complete all of the requirements.");
87     Console.ResetColor();
88     Console.Write("Answer: ");
89     answers[5] = Convert.ToInt32(Console.ReadLine());
90
91     Console.ForegroundColor = ConsoleColor.Green;
92     Console.WriteLine("\nI was unable to maintain a healthy study routine.");
93     Console.ResetColor();
94     Console.Write("Answer: ");
95     answers[6] = Convert.ToInt32(Console.ReadLine());
96
97     Console.ForegroundColor = ConsoleColor.Green;
98     Console.WriteLine("\nBecause the online course offers a lot of information, it affects my mental state and does not improve my memory.");
99     Console.ResetColor();
100    Console.Write("Answer: ");
101    answers[7] = Convert.ToInt32(Console.ReadLine());
102
103    Console.ForegroundColor = ConsoleColor.Green;
104    Console.WriteLine("\nMy self-discipline worsened, and my time management suffered as a result.");
105    Console.ResetColor();
106    Console.Write("Answer: ");
107    answers[8] = Convert.ToInt32(Console.ReadLine());
108
109    Console.ForegroundColor = ConsoleColor.Green;
110    Console.WriteLine("\nI get frustrated when the internet is slow, and as a result, I get stressed and upset.");
111    Console.ResetColor();
112    Console.Write("Answer: ");
113    answers[9] = Convert.ToInt32(Console.ReadLine());
114
115    // below is to show the result of the survey
116    Console.ForegroundColor = ConsoleColor.Green;
117    Console.WriteLine("\nResult:");
118    Console.ResetColor();
119
120

```

Figure 8. The method for Answer Survey

The first option that the surveyee may use is answering the survey, This method includes getting the information from the user such as name, age, sex, and course. After that, the directions and likert-scale questions will appear.



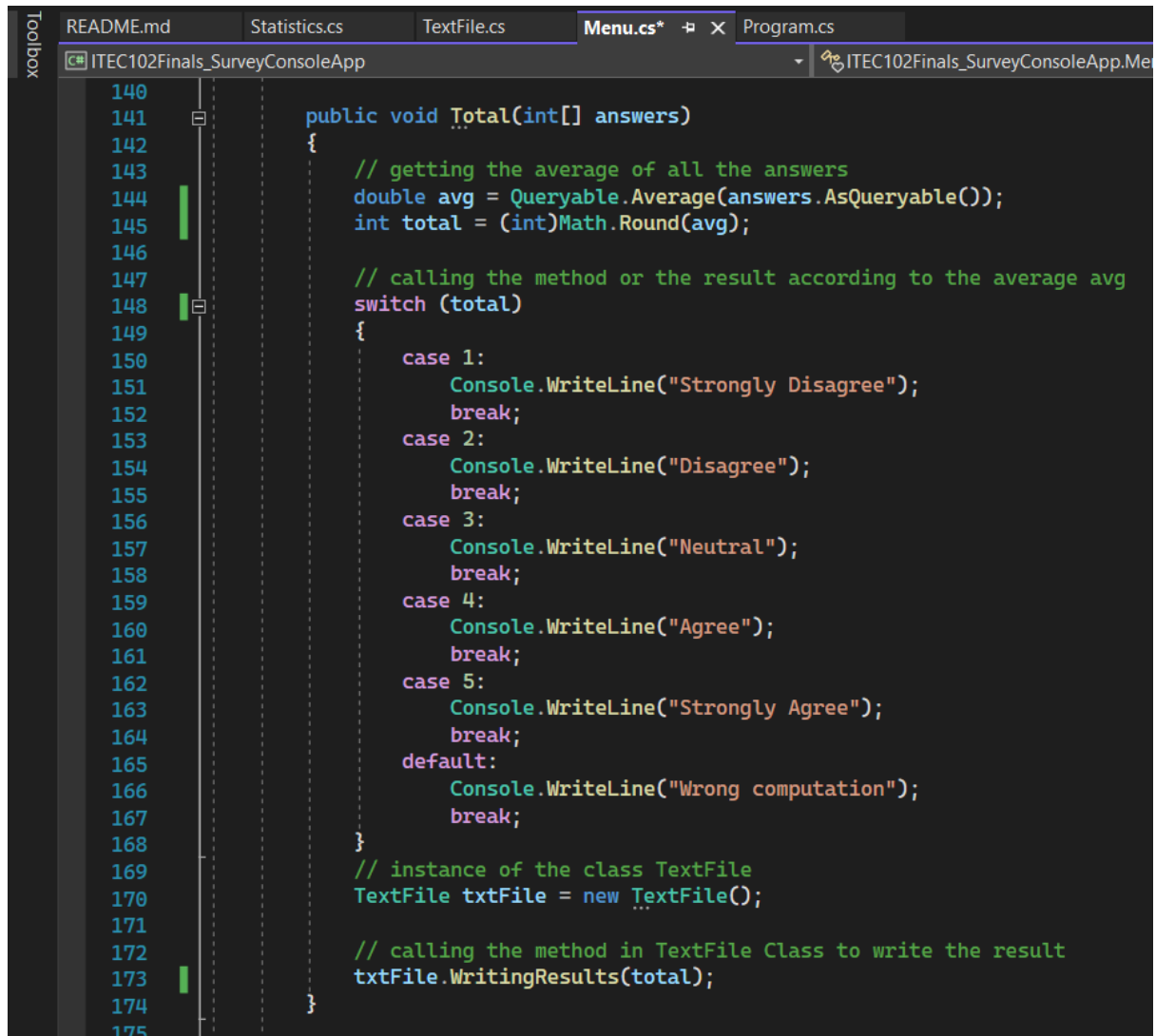
```

115 // below is to show the result of the survey
116 Console.ForegroundColor = ConsoleColor.Green;
117 Console.WriteLine("\n=====");
118 Console.WriteLine("Result");
119 Console.ResetColor();
120
121 // calling the method inside the same class to calculate to avg
122 Total(answers);
123 Console.ForegroundColor = ConsoleColor.Green;
124 Console.WriteLine("=====");
125 Console.ResetColor();
126
127 // instance of the class TextFile
128 TextFile txtFile = new TextFile();
129
130 // calling the methods inside the TextFile class to write the surveyee's info right away
131 txtFile.WritingInfo(name, age, sex, course, answers);
132 txtFile.WritingAge(age);
133 txtFile.WritingSex(sex);
134 txtFile.WritingCourse(course);
135
136 Console.ForegroundColor = ConsoleColor.DarkGreen;
137 Console.WriteLine("\nThank you for answering!");
138 Console.ResetColor();
139 }
140

```

Figure 9. Calling the methods from TextFile class

After the user answers all the 10 questions, the result will be displayed which is also based on the Total method. Following, the methods from the TextFile class will be called. Then, the "Thank you for answering!" will be shown and then the program will take the user back to the menu options.



```
140
141 public void Total(int[] answers)
142 {
143     // getting the average of all the answers
144     double avg = Queryable.Average(answers.AsQueryable());
145     int total = (int)Math.Round(avg);
146
147     // calling the method or the result according to the average avg
148     switch (total)
149     {
150         case 1:
151             Console.WriteLine("Strongly Disagree");
152             break;
153         case 2:
154             Console.WriteLine("Disagree");
155             break;
156         case 3:
157             Console.WriteLine("Neutral");
158             break;
159         case 4:
160             Console.WriteLine("Agree");
161             break;
162         case 5:
163             Console.WriteLine("Strongly Agree");
164             break;
165         default:
166             Console.WriteLine("Wrong computation");
167             break;
168     }
169     // instance of the class TextFile
170     TextFile txtFile = new TextFile();
171
172     // calling the method in TextFile Class to write the result
173     txtFile.WritingResults(total);
174 }
175
```

Figure 10. Method for getting the equivalent of total

This is where the average of all the answers will get. Then, based on it, it will call the method or the result according to that average. Next, an instantiation of a class called TextFile, which has a method called WritingResults is called. The text file will be writing out what it gets as its total value. The code will call the method or the result according to the average.

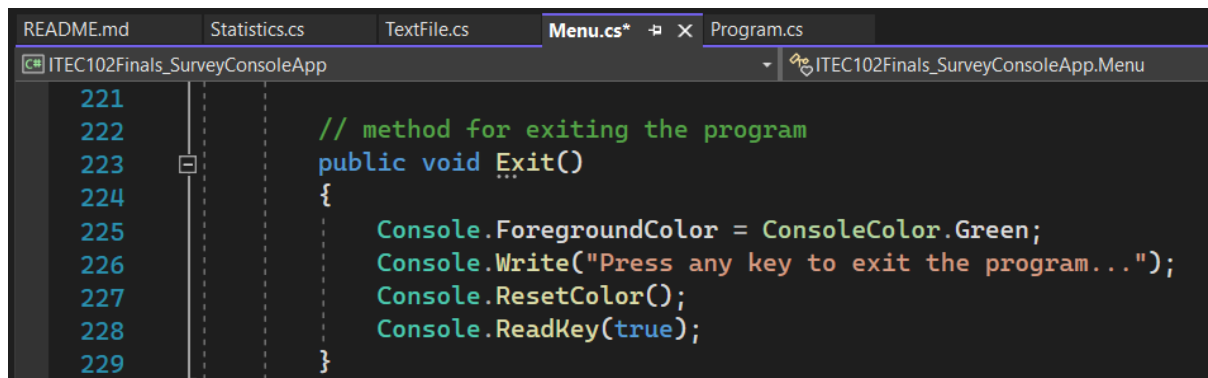
```

176 // to show less statistics in the console
177 public void ShowCurrentStats()
178 {
179     Console.ForegroundColor = ConsoleColor.Green;
180     Console.WriteLine("\n=====");
181     Console.WriteLine("Survey's Statistics");
182     Console.ResetColor();
183
184     // getting the number of surveyees by reading the length of ages wrote in its textfile
185     string[] ageStats = File.ReadAllLines(@"D:\ITEC102Finals\survey - age.txt");
186     int surveyees = ageStats.Length;
187
188     // instance of the class Statistics
189     Statistics stats = new Statistics();
190
191     // read the number of surveyee starting from the start in text file
192     Console.WriteLine($"Total number of surveyee: {surveyees}");
193
194     // calling the methods in the Statistics class
195     Console.ForegroundColor = ConsoleColor.Green;
196     Console.WriteLine("\nAge");
197     Console.ResetColor();
198     stats.ReadingAge();
199
200     Console.ForegroundColor = ConsoleColor.Green;
201     Console.WriteLine("\nSex");
202     Console.ResetColor();
203     stats.ReadingSex();
204
205     Console.ForegroundColor = ConsoleColor.Green;
206     Console.WriteLine("\nCourse");
207     Console.ResetColor();
208     stats.ReadingCourse();
209
210     Console.ForegroundColor = ConsoleColor.Green;
211     Console.WriteLine("\nResults");
212     Console.ResetColor();
213     stats.ReadingResults();
214
215     stats.WritingStats();
216
217     Console.ForegroundColor = ConsoleColor.Green;
218     Console.WriteLine("\n=====");
219     Console.ResetColor();
220 }
221

```

Figure 11. The method for showing the current statistics

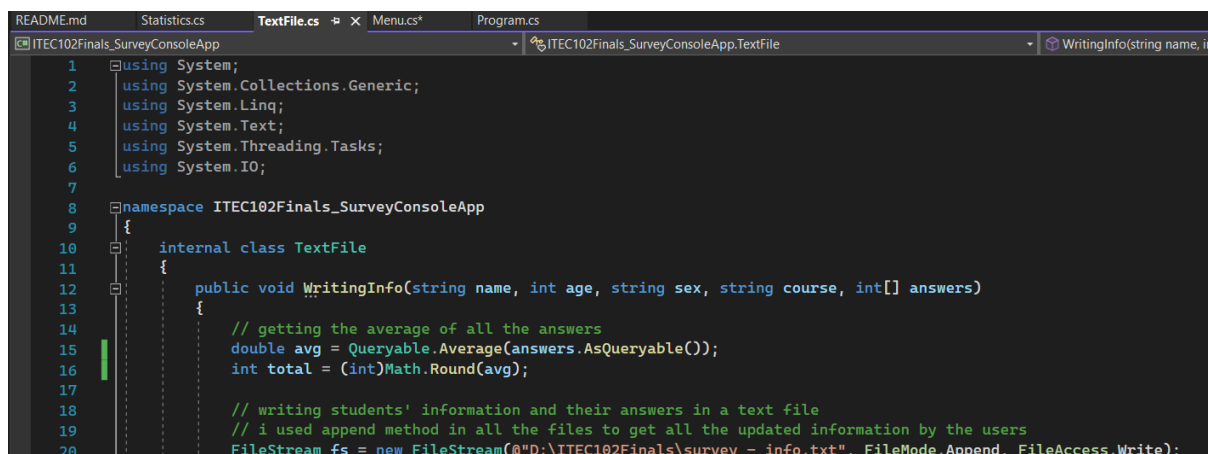
The code attempts to show the survey's statistics. It gets the total number of surveyees by reading the length of ages written in its text file called "survey - age". Next, it shows the statistics of surveyees' age, sex, course, results by reading from its each own text files.



```
221
222 // method for exiting the program
223 public void Exit()
224 {
225     Console.ForegroundColor = ConsoleColor.Green;
226     Console.WriteLine("Press any key to exit the program...");
227     Console.ResetColor();
228     Console.ReadKey(true);
229 }
```

Figure 12. The method for exiting the program

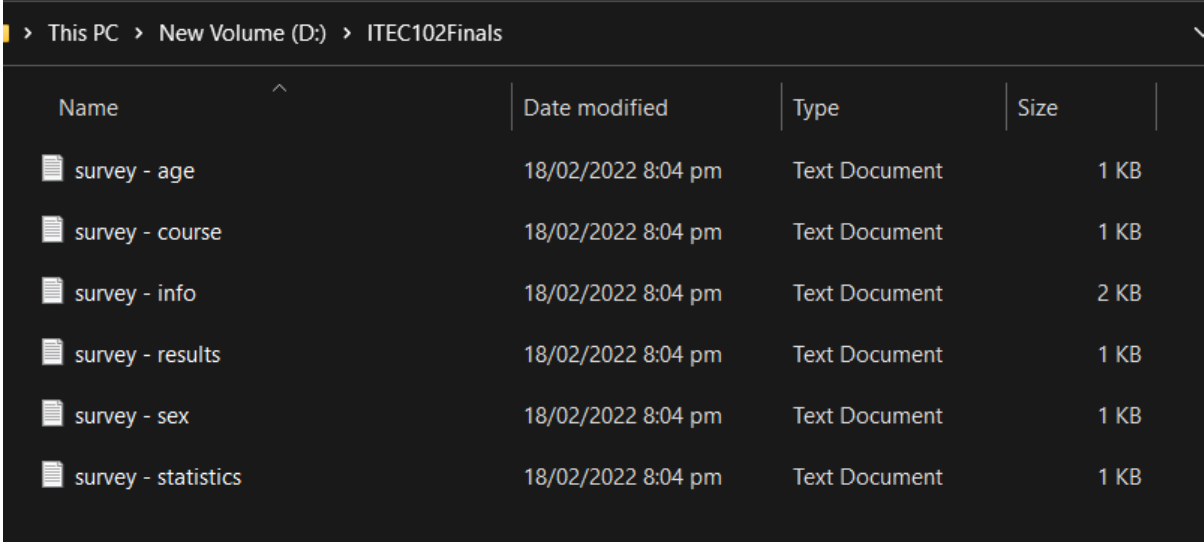
This is the method that will be called when the user chooses 'E' from the Menu option. The code block contains instructions that the program will be closed when the user pressed any key.



```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using System.IO;
7
8 namespace ITEC102Finals_SurveyConsoleApp
9 {
10     internal class TextFile
11     {
12         public void WritingInfo(string name, int age, string sex, string course, int[] answers)
13         {
14             // getting the average of all the answers
15             double avg = Queryable.Average(answers.AsQueryable());
16             int total = (int)Math.Round(avg);
17
18             // writing students' information and their answers in a text file
19             // i used append method in all the files to get all the updated information by the users
20             FileStream fs = new FileStream(@"D:\ITEC102Finals\survey - info.txt", FileMode.Append, FileAccess.Write);
```

Figure 13. The TextFile class

This class just includes all the methods to write in a text file. One of the main benefits of storing your data somewhere is protecting your files from data thieves and unwanted guests, and also to have it saved digitally (Capital Business Solutions, 2016). The reason why the researchers store it in a text file is because that is the one they have learned from first semester and also to have it stored somewhere where it can also retrieve the information from previous surveys since it is essential for statistical analysis.



Name	Date modified	Type	Size
survey - age	18/02/2022 8:04 pm	Text Document	1 KB
survey - course	18/02/2022 8:04 pm	Text Document	1 KB
survey - info	18/02/2022 8:04 pm	Text Document	2 KB
survey - results	18/02/2022 8:04 pm	Text Document	1 KB
survey - sex	18/02/2022 8:04 pm	Text Document	1 KB
survey - statistics	18/02/2022 8:04 pm	Text Document	1 KB

Figure 14. *The text files from the program*

These are the text files created by the program which is saved in the “ITEC102Finals” folder in the D drive. The “survey - info” has the information of the students and their answers to the questions, the “survey - statistics” has the current statistics, and the rest of the text files have just the data of the surveyees from each of it.

```

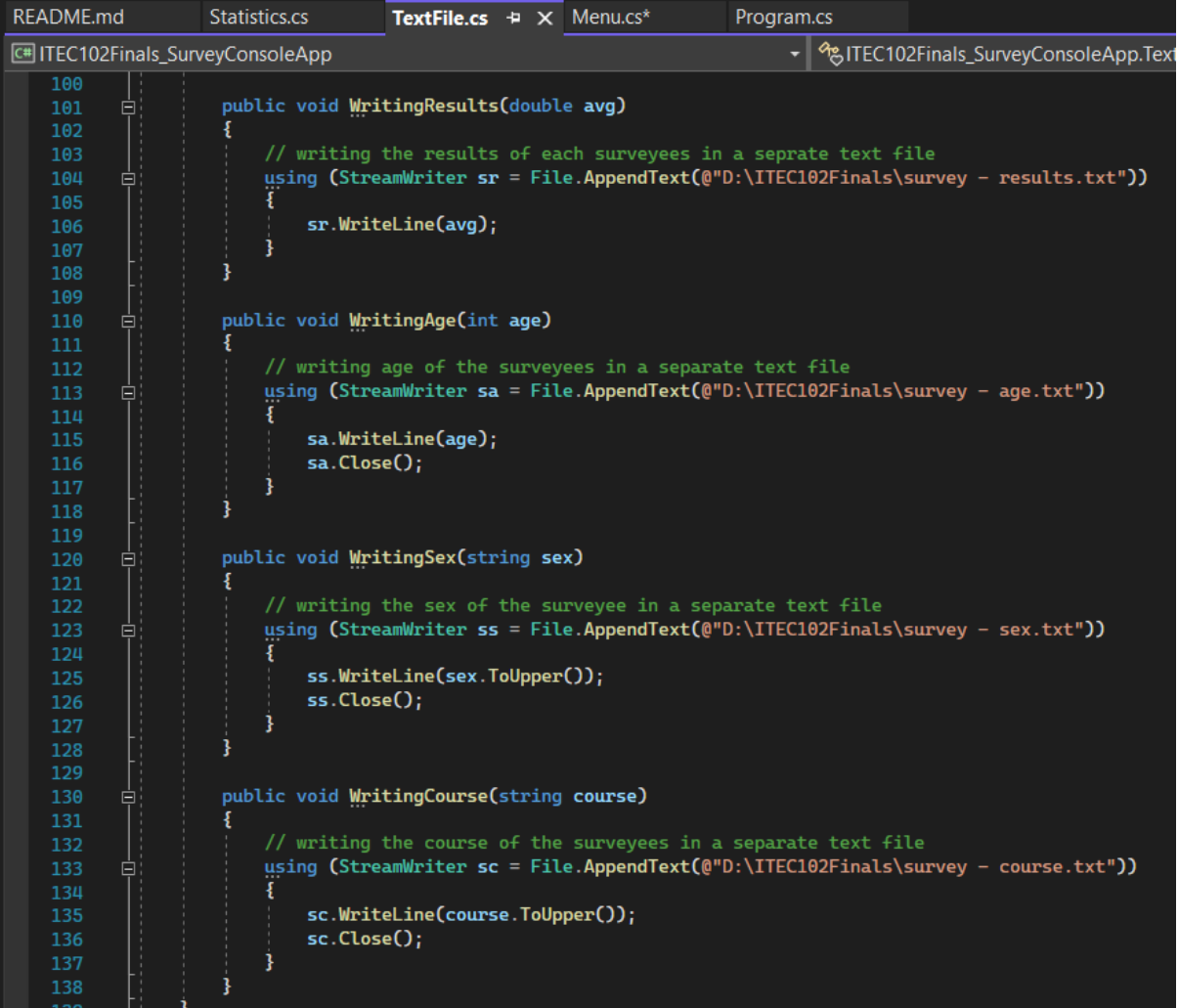
11 public void WritingInfo(string name, int age, string sex, string course, int[] answers)
12 {
13     // getting the average of all the answers
14     double avg = Queryable.Average(answers.AsQueryable());
15     int total = (int)Math.Round(avg);
16
17     // writing students' information and their answers in a text file
18     // i used append method in all the files to get all the updated information by the users
19     FileStream fs = new FileStream(@"D:\ITEC102Finals\survey - info.txt", FileMode.Append, FileAccess.Write);
20     using (StreamWriter sw = new StreamWriter(fs))
21     {
22         // this is to get the place number of surveyee and appends it
23         int num = 0;
24         if (File.Exists(@"D:\ITEC102Finals\survey - age.txt"))
25         {
26             string[] surveyee = File.ReadAllLines(@"D:\ITEC102Finals\survey - age.txt");
27             num = surveyee.Length;
28         }
29
30         sw.WriteLine($"Surveyee #{num + 1}"); // to show what number the surveyee is
31         // info of surveyee
32         sw.WriteLine($"Name : {name}");
33         sw.WriteLine($"Age : {age}");
34         sw.WriteLine($"Sex : {sex}");
35         sw.WriteLine($"Course : {course}");
36
37         // answers of surveyee
38         sw.WriteLine("Answers:");
39         foreach (int ans in answers)
40         {
41             // so the int ans will be converted into equivalent string
42             string ansString;
43             switch (ans)
44             {
45                 case 1:
46                     ansString = "Strongly Disagree";
47                     sw.WriteLine(" " + ansString);
48                     break;
49                 case 2:
50                     ansString = "Disagree";
51                     sw.WriteLine(" " + ansString);
52                     break;
53                 case 3:
54                     ansString = "Neutral";
55                     sw.WriteLine(" " + ansString);
56                     break;
57                 case 4:
58                     ansString = "Agree";
59                     sw.WriteLine(" " + ansString);
60                     break;
61                 case 5:
62                     ansString = "Strongly Agree";
63                     sw.WriteLine(" " + ansString);
64                     break;
65                 default:
66                     sw.WriteLine(" Wrong output");
67                     break;
68             }
69         }
70
71         // writing the result or the average of the surveyees' answers
72         sw.WriteLine("\nResult : ");
73         switch (total)
74         {
75             case 1:
76                 sw.WriteLine("Strongly Disagree");
77                 break;
78             case 2:
79                 sw.WriteLine("Disagree");
80                 break;
81             case 3:
82                 sw.WriteLine("Neutral");
83                 break;
84             case 4:
85                 sw.WriteLine("Agree");
86                 break;
87             case 5:
88                 sw.WriteLine("Strongly Agree");
89                 break;
90             default:
91                 sw.WriteLine("Wrong output");
92                 break;
93         }
94         sw.WriteLine("\n=====");
95     }
96     sw.Close();

```

Figure 15. The method for creating “survey-info”

The code starts by getting the average of all the answers. Then, it calculates how many students are in the class and their average score. It then writes out all the information about each student to a text file using the append method. This part to write students'

information and their answers in a text file. Above uses the append method in all the files to get all the updated information by the users.



```
100
101 public void WritingResults(double avg)
102 {
103     // writing the results of each surveyees in a seprate text file
104     using (StreamWriter sr = File.AppendText(@"D:\ITEC102Finals\survey - results.txt"))
105     {
106         sr.WriteLine(avg);
107     }
108 }
109
110 public void WritingAge(int age)
111 {
112     // writing age of the surveyees in a separate text file
113     using (StreamWriter sa = File.AppendText(@"D:\ITEC102Finals\survey - age.txt"))
114     {
115         sa.WriteLine(age);
116         sa.Close();
117     }
118 }
119
120 public void WritingSex(string sex)
121 {
122     // writing the sex of the surveyee in a separate text file
123     using (StreamWriter ss = File.AppendText(@"D:\ITEC102Finals\survey - sex.txt"))
124     {
125         ss.WriteLine(sex.ToUpper());
126         ss.Close();
127     }
128 }
129
130 public void WritingCourse(string course)
131 {
132     // writing the course of the surveyees in a separate text file
133     using (StreamWriter sc = File.AppendText(@"D:\ITEC102Finals\survey - course.txt"))
134     {
135         sc.WriteLine(course.ToUpper());
136         sc.Close();
137     }
138 }
139 }
```

Figure 16. *The methods for writing other data to a text file*

This part writes the results, age, sex, and course of surveyees in a separate text file. It uses StreamWriter class and File class to create a new text file with the name "survey - (data).txt" on a folder named "ITEC102Finals" in drive D:.


```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Text.RegularExpressions;
6  using System.Threading.Tasks;
7  using System.IO;
8
9  namespace ITEC102Finals_SurveyConsoleApp
10 {
11     internal class Statistics
12     {
13         int eighteen;
14         int nineteen;
15         int twenty;
16
17         int female;
18         int male;
19
20         int bscs;
21         int bsis;
22         int bsit;
23
24         int stronglyAgree;
25         int agree;
26         int neutral;
27         int disagree;
28         int stronglyDisagree;
29
30         string[] arrayAge = File.ReadAllLines(@"D:\ITEC102Finals\survey - age.txt");
31

```

Figure 17. The Statistics class

This class where the code for getting the statistics will be written. The variables are here so the last method which is writing the statistics in a text file can access it.

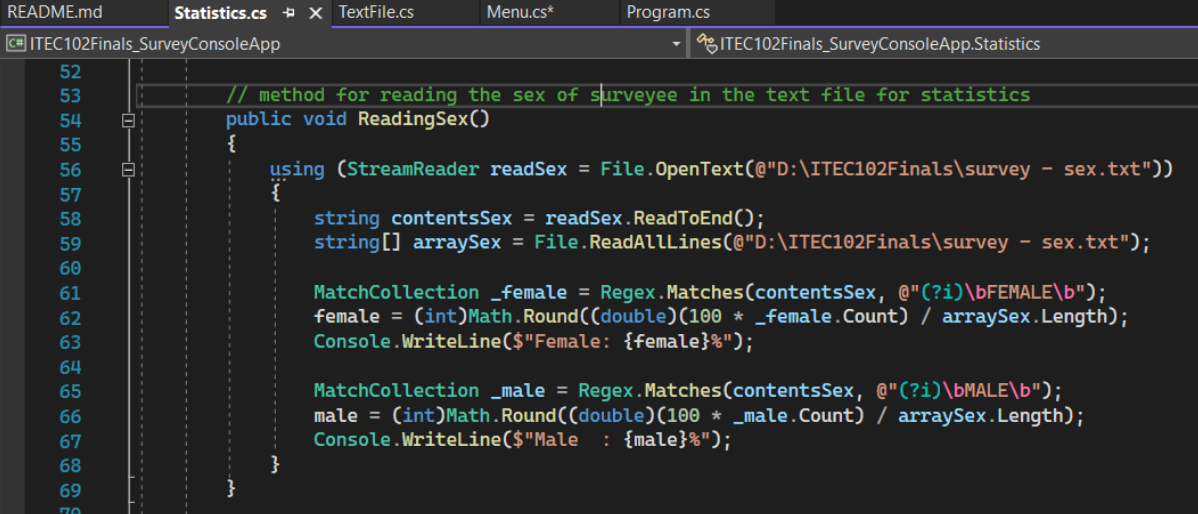
```

31
32 // method for reading the age in the text file to show the statistics
33 public void ReadingAge()
34 {
35     using (StreamReader readAge = new StreamReader(@"D:\ITEC102Finals\survey - age.txt"))
36     {
37         string contentsAge = readAge.ReadToEnd();
38
39         MatchCollection _eighteen = Regex.Matches(contentsAge, @"(?i)\b18\b");
40         eighteen = (int)Math.Round((double)(100 * _eighteen.Count) / arrayAge.Length);
41         Console.WriteLine($"18: {eighteen}%");
42
43         MatchCollection _nineteen = Regex.Matches(contentsAge, @"(?i)\b19\b");
44         nineteen = (int)Math.Round((double)(100 * _nineteen.Count) / arrayAge.Length);
45         Console.WriteLine($"19: {nineteen}%");
46
47         MatchCollection _twenty = Regex.Matches(contentsAge, @"(?i)\b20\b");
48         twenty = (int)Math.Round((double)(100 * _twenty.Count) / arrayAge.Length);
49         Console.WriteLine($"20: {twenty}%");
50     }
51 }
52

```

Figure 18. Statistics of Age

This part is used to read the age of each surveyee. It starts by using a StreamReader to read the text file "survey - age.txt" into memory. It then uses a string variable called contentAge and reads it until the end of the file. The next step is to use regular expressions to match all lines that contain "18", "19", "20" in them and store those matches in their own array called _eighteen, _nineteen, _twenty. Then, use computation to get the percentage and store it in another array: eighteen, nineteen, twenty. Finally, print it.



```
52
53 // method for reading the sex of surveyee in the text file for statistics
54 public void ReadingSex()
55 {
56     using (StreamReader readSex = File.OpenText(@"D:\ITEC102Finals\survey - sex.txt"))
57     {
58         string contentsSex = readSex.ReadToEnd();
59         string[] arraySex = File.ReadAllLines(@"D:\ITEC102Finals\survey - sex.txt");
60
61         MatchCollection _female = Regex.Matches(contentsSex, @"(?i)\bFEMALE\b");
62         female = (int)Math.Round((double)(100 * _female.Count) / arraySex.Length);
63         Console.WriteLine($"Female: {female}%");
64
65         MatchCollection _male = Regex.Matches(contentsSex, @"(?i)\bMALE\b");
66         male = (int)Math.Round((double)(100 * _male.Count) / arraySex.Length);
67         Console.WriteLine($"Male : {male}%");
68     }
69 }
70
```

Figure 19. Statistics of Sex

This part is used to read the sex of each surveyee. It starts by using a StreamReader to read the text file "survey - sex.txt" into memory. It then uses a string variable called contentSex and reads it until the end of the file. The next step is to use regular expressions to match all lines that contain "FEMALE", "MALE" in them and store those matches in their own array called _female, _male. Then, use computation to get the percentage and store it in another array: female, male. Finally, print it.

```

70
71 // method for reading the course of surveyee inside the text file for statistics
72 public void ReadingCourse()
73 {
74     using (StreamReader readCourse = File.OpenText(@"D:\ITEC102Finals\survey - course.txt"))
75     {
76         string contentsCourse = readCourse.ReadToEnd();
77         string[] arrayCourse = File.ReadAllLines(@"D:\ITEC102Finals\survey - course.txt");
78
79         MatchCollection _bscs = Regex.Matches(contentsCourse, @"(?:)\bBSCS\b");
80         bscs = (int)Math.Round((double)(100 * _bscs.Count) / arrayCourse.Length);
81         Console.WriteLine($"BSCS: {bscs}%");
82
83         MatchCollection _bsis = Regex.Matches(contentsCourse, @"(?:)\bBSIS\b");
84         bsis = (int)Math.Round((double)(100 * _bsis.Count) / arrayCourse.Length);
85         Console.WriteLine($"BSIS: {bsis}%");
86
87         MatchCollection _bsit = Regex.Matches(contentsCourse, @"(?:)\bBSIT\b");
88         bsit = (int)Math.Round((double)(100 * _bsit.Count) / arrayCourse.Length);
89         Console.WriteLine($"BSIT: {bsit}%");
90     }
91 }
92

```

Figure 20. Statistics of Course

This part is used to read the course of each surveyee. It starts by using a StreamReader to read the text file "survey - course.txt" into memory. It then uses a string variable called contentCourse and reads it until the end of the file. The next step is to use regular expressions to match all lines that contain "BSCS", "BSIT", "BSIS" in them and store those matches in their own array called _bscs, _bsit, _bsis. Then, use computation to get the percentage and store it in another array: bscs, bsit, bsis. Finally, print it.

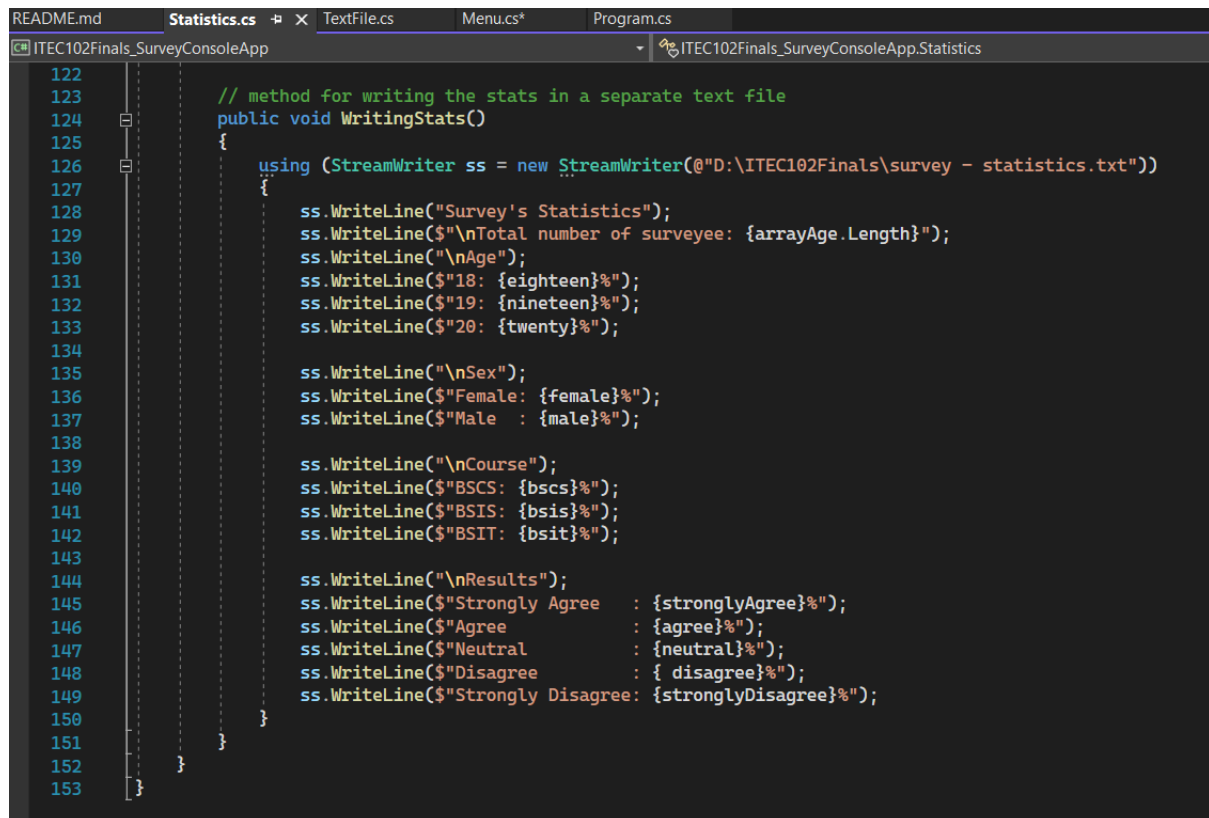
```

92
93 // method for reading the results of the surveyees inside the text file for statistics
94 public void ReadingResults()
95 {
96     using (StreamReader readResults = File.OpenText(@"D:\ITEC102Finals\survey - results.txt"))
97     {
98         string contentResults = readResults.ReadToEnd();
99         string[] arrayResults = File.ReadAllLines(@"D:\ITEC102Finals\survey - results.txt");
100
101         MatchCollection _stronglyAgree = Regex.Matches(contentResults, @"(?:)\b5\b");
102         stronglyAgree = (int)Math.Round((double)(100 * _stronglyAgree.Count) / arrayResults.Length);
103         Console.WriteLine($"Strongly Agree : {stronglyAgree}%");
104
105         MatchCollection _agree = Regex.Matches(contentResults, @"(?:)\b4\b");
106         agree = (int)Math.Round((double)(100 * _agree.Count) / arrayResults.Length);
107         Console.WriteLine($"Agree : {agree}%");
108
109         MatchCollection _neutral = Regex.Matches(contentResults, @"(?:)\b3\b");
110         neutral = (int)Math.Round((double)(100 * _neutral.Count) / arrayResults.Length);
111         Console.WriteLine($"Neutral : {neutral}%");
112
113         MatchCollection _disagree = Regex.Matches(contentResults, @"(?:)\b2\b");
114         disagree = (int)Math.Round((double)(100 * _disagree.Count) / arrayResults.Length);
115         Console.WriteLine($"Disagree : {disagree}%");
116
117         MatchCollection _stronglyDisagree = Regex.Matches(contentResults, @"(?:)\b1\b");
118         stronglyDisagree = (int)Math.Round((double)(100 * _stronglyDisagree.Count) / arrayResults.Length);
119         Console.WriteLine($"Strongly Disagree: {stronglyDisagree}%");
120     }
121 }
122

```

Figure 21. Statistics of Results

This part is used to read the results of each surveyee. It starts by using a StreamReader to read the text file "survey - results.txt" into memory. It then uses a string variable called contentResults and reads it until the end of the file. The next step is to use regular expressions to match all lines that contain "5", "4", "3", "2", "1" in them and store those matches in their own array called _stronglyAgree, _agree, etc. Then, use computation to get the percentage and store it in another array: stronglyAgree, agree, etc. Finally, print it.



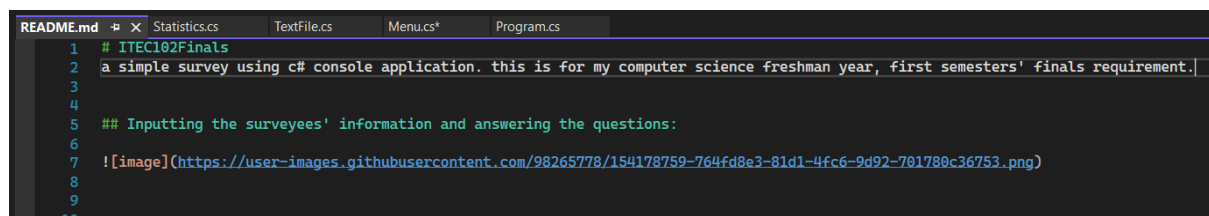
```

122 // method for writing the stats in a separate text file
123 public void WritingStats()
124 {
125     using (StreamWriter ss = new StreamWriter(@"D:\ITEC102Finals\survey - statistics.txt"))
126     {
127         ss.WriteLine("Survey's Statistics");
128         ss.WriteLine($"Total number of surveyee: {arrayAge.Length}");
129         ss.WriteLine("\nAge");
130         ss.WriteLine($"18: {eighteen}%");
131         ss.WriteLine($"19: {nineteen}%");
132         ss.WriteLine($"20: {twenty}%");
133
134         ss.WriteLine("\nSex");
135         ss.WriteLine($"Female: {female}%");
136         ss.WriteLine($"Male : {male}%");
137
138         ss.WriteLine("\nCourse");
139         ss.WriteLine($"BSCS: {bscs}%");
140         ss.WriteLine($"BSIS: {bsis}%");
141         ss.WriteLine($"BSIT: {bsit}%");
142
143         ss.WriteLine("\nResults");
144         ss.WriteLine($"Strongly Agree : {stronglyAgree}%");
145         ss.WriteLine($"Agree : {agree}%");
146         ss.WriteLine($"Neutral : {neutral}%");
147         ss.WriteLine($"Disagree : {disagree}%");
148         ss.WriteLine($"Strongly Disagree: {stronglyDisagree}%");
149     }
150 }
151
152
153

```

Figure 22. Writing the overall statistics in a text file

The code starts by using a StreamWriter object to write the text "Survey's Statistics" to a file called D:\ITEC 102 Finals\survey - statistics.txt. Next, it writes out the number of surveyees in the arrayAge variable and their age, sex, course and results. It's the same format as the one in the showCurrentStatistics method from the Menu class.



```

1 # ITEC102Finals
2 a simple survey using c# console application. this is for my computer science freshman year, first semesters' finals requirement.
3
4
5 ## Inputting the surveyees' information and answering the questions:
6
7 ![image](https://user-images.githubusercontent.com/98265778/154178759-764fd8e3-81d1-4fc6-9d92-701780c36753.png)
8
9
10

```

Figure 23. README.md for GitHub

This file is just for GitHub purposes. It is important since Readme provides an introduction to the files contained in the repository and they prevent a person viewing or using your repository from needing to read your mind (Kadish, 2020),

III. Conclusions and Recommendations

This research has looked at mental health and its impact on first-year college students. It's either that students can't attend online meetings due to mental health issues, or that it causes burnout.

A. Summary

The educational system has suddenly been hit by an unexpected health crisis, which has fractured its foundation. Shown that online learning challenges among college students, particularly freshmen, varied in terms of mental health well-being, that they may have experienced stress, anxiety, and depression. Their greatest concerning issue was their home learning environment, while technical knowledge and expertise was their least pressing concern. So, using a C# console application, the researchers created a survey to see how many freshmen students from College of Computer Studies are having this issue.

B. Significance of the Project

This study will benefit the following:

STUDENTS. This study will provide an insight at how first-year college students deal with stress, how anxiety affects their academic performance, and how they feel about online learning. This project will benefit students, particularly programming students, by demonstrating how to conduct surveys using a console application.

TEACHERS. The project's findings will assist instructors/teachers in understanding what students are going through in online classes and how they may adapt or recognize when they need to cut back on their schoolwork. Since they're the second parent, they can also guide or assist students who are experiencing stress or anxiety.

FUTURE RESEARCHER. The concepts offered could be used as a starting point for fresh research, and it will also benefit them because they will understand how to use the C# console software to perform surveys and other tasks. This study also demonstrates the benefits of exploring what console applications can do. It will also act as a guide and provide background information on the subject.

C. Recommendations

This research has contributed to a greater understanding of the issues that students face while they learn in a hybrid online environment. As the investigation progressed, a few areas emerged as potential future research areas. The recommendation are as follows:

- a) Given the lockdown, one of the primary concerns is the students' mental health. Research has shown that online learning issues among college students, particularly freshmen, vary in terms of mental health well-being, with some reporting stress, anxiety, and depression. It should prioritize the study's ability to contribute to a better understanding of health disparities among college students, including the formation of a working group to oversee this area and the support of console app research.
- b) Mental Illness exists, from a small problem it will grow eventually (Reyes, 2021). It may worsen for those who already have mental health issues. It's more difficult for students to learn new knowledge, even if they merely sit in front of the computer.
- c) This study needs to create a clearer justification of its findings concerning factors that can enhance exploratory thinking. These should concentrate on

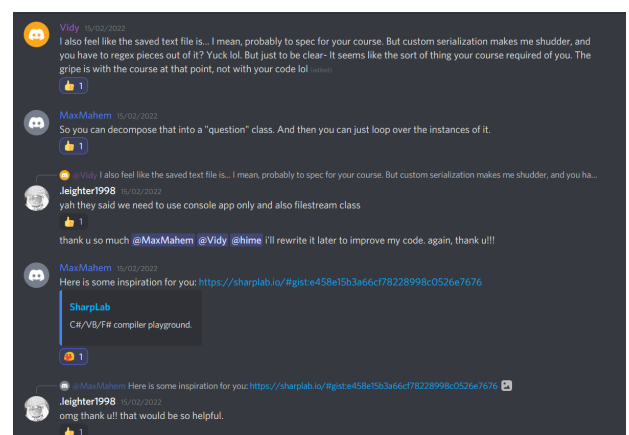
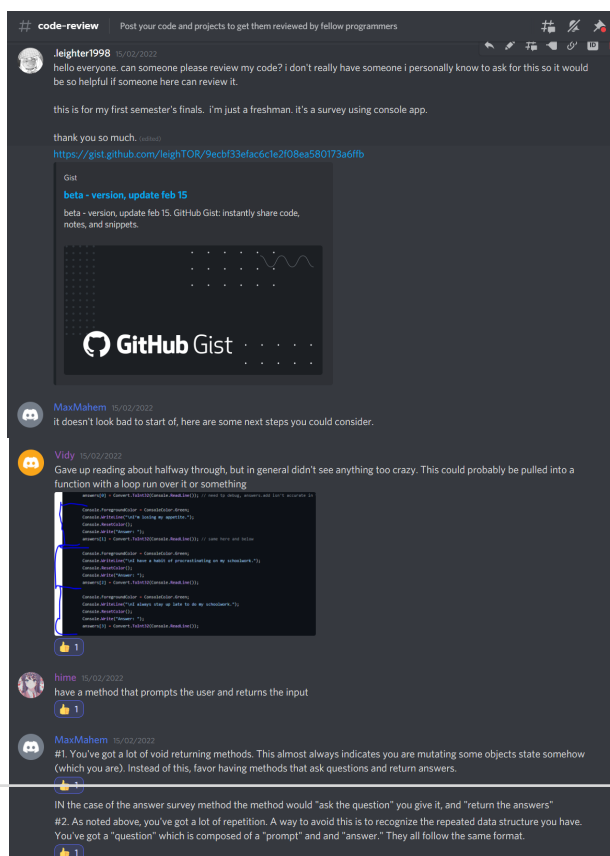
future console app code or programs.

IV. Future Work

The researchers did all they can in those 2 weeks of doing this study but of course there are still things that need to be improved. Since the researchers don't have enough time to make the study great, below are some aspects that may follow so that the program and the study may improve to the better version of itself. If anyone who read this study wants to improve the code, they can fork it through GitHub, here is the link: <https://github.com/leighTOR/survey-console.git>

Here are some things to follow:

- Improve the code by making it more efficient.
- Since the study was based on the current knowledge of the researchers, the code may improve by attaining the intermediate level of programming, or better, the advanced level.
- Follow the advice of senior developers from Discord to improve codes. Below are some comments and suggestions from them:



References

- Baltà-Salvador, R., Olmedo-Torre, N., & Peña, M. (2021, June 05). *Academic and emotional effects of online learning during the COVID-19 pandemic on engineering students*.
<https://doi.org/10.1007/s10639-021-10593-1>
- Bruffaerts, R., Mortier, P., Kiekens, G., & Auerbach, R. (2018). *Mental health problems in college freshmen: Prevalence and academic functioning*. Journal of Affective Disorders. <https://www.sciencedirect.com/science/article/pii/S0165032716324545>
- Capital Business Solutions. (2016, October 21). *The Benefits of Storing Information Digitally*. Capital Business Systems. Retrieved February 19, 2022, from <https://www.capitalmds.com/benefits-storing-information-digitally/>
- Ganesh, A. (2019, June 7). *Encapsulation In C#*. C# Corner. Retrieved February 18, 2022, from <https://www.c-sharpcorner.com/article/encapsulation-in-C-Sharp/>
- Kadish, C. (2020, November 17). *README files are important. PURPOSE: To provide guidance and...* |. Towards Data Science. Retrieved February 19, 2022, from <https://towardsdatascience.com/dear-student-readme-files-are-important-8e4773035924>
- Mental Health*. (2021, September 1). MedlinePlus. Retrieved February 19, 2022, from <https://medlineplus.gov/mentalhealth.html>
- Reyes, I. T. (2021, July 22). *Online Classes in the Philippines. Is Mental Health more important than Education?* Medium. Retrieved February 20, 2022, from <https://medium.com/@hexisdylan23/online-classes-in-the-philippines-is-mental-health-more-important-than-education-fe30bcf702e4>