**COMP 1202**

**Assignment 1**

**Due: Sunday, February 16, 2025 (11:00 p.m.)**

**Given: Sunday, January 29, 2025**

Student Name: (as per Stu-view) \_\_Jui Hsin Wong\_\_\_\_

Student Number: \_101559700\_\_

**Submission Requirements:**

**You must submit 2 Documents:**

* You are required to submit the source code (.cs file)
* You must also paste you complete code into this document and upload the screenshots of your program output and interaction

**You MUST:**

* **Clearly comment your code (At least with your name and id number).**
* **You must use appropriate variable names.**
* **Print meaningful input and output statements.**
* **Indent you code where necessary.**

**All submissions should at least compile including non-working submissions.**

**Code:**

namespace comp\_1201\_Object\_Oriented\_Programming1;

public class assignment1

{

static string AssignmentMenu()

{

Console.WriteLine(("--Main Menu--"));

Console.WriteLine(("1. Display information"));

Console.WriteLine(("2. Manage date and time"));

Console.WriteLine(("3. Handle text "));

Console.WriteLine(("4. Exit the program"));

Console.WriteLine(("-------------"));

Console.WriteLine(("Enter 1, 2, 3 or 4"));

return Console.ReadLine(); // collect this in a form of string!

}

public static void MainMenu()

{

while (true)

{

var choice = AssignmentMenu();

if (choice == "1") { Information(); }

else if (choice == "2") { DateTimeCalculate(); }

else if (choice == "3") { HandleText(); }

else if (choice == "4") { Console.WriteLine(("Exit the program")); break; }

else

{

Console.WriteLine("Bad input, please try again");

}

}

}

public static string InformationMenu()

{

Console.WriteLine("------Display information-------");

Console.WriteLine("Name: Jui Hsin Wong");

Console.WriteLine("ID: 101559700");

Console.WriteLine("=================");

Console.WriteLine("Enter 1 go to previous menu");

return Console.ReadLine(); ;

}

public static void Information(){

while(true){

var choice = InformationMenu();

if (choice=="1"){Console.WriteLine("Go back to previous menu");break;}

else

{

Console.WriteLine("Bad input, please try again");

}

}

}

private static string DateTimeMenu()

{

Console.WriteLine("------Date and Time Operations Menu-------");

Console.WriteLine("1- Display Current Date and Time");

Console.WriteLine("2- Add Days to Current Date");

Console.WriteLine("3- Calculate Age from your birthdate");

Console.WriteLine("4- Go back to previous menu");

Console.WriteLine("=================");

Console.WriteLine("Enter 1,2,3 or 4");

return Console.ReadLine();

}

private static void DateTimeCalculate()

{

while (true)

{

var choice = DateTimeMenu();

DateTime now = DateTime.Now;

if (choice == "1" || choice == "2" || choice == "3")

{

if (choice == "1") Console.WriteLine($"The current time and date is {now}");

else if (choice == "2")

{

Console.WriteLine("How many Days you want to add?");

string input = Console.ReadLine();

//var AddDate = int.Parse(Console.ReadLine());

if (int.TryParse(input, out int AddDate))

{

Console.WriteLine($"The date afrer adding {AddDate} days is {now.AddDays(AddDate)}");

}

else {

Console.WriteLine("Bad input, please try again");

}

}

else if (choice == "3")

{

Console.WriteLine("What is your birthday(yyyy/mm/dd)");

if (DateTime.TryParse(Console.ReadLine(), out DateTime BirthDate))

{

DateTime now1 = DateTime.Now;

int age = now1.Year - BirthDate.Year;

if (now1 < BirthDate.AddYears(age))

{

age--;

}

Console.WriteLine($"Your age is {age}");

}

else

{

Console.WriteLine("You did not enter a valid date");

}

}

}

else if (choice == "4") { Console.Clear(); return; }

else {

Console.WriteLine("Bad input, please try again");

}

}

}

private static string HandleTextMenu()

{

Console.WriteLine("------ Text Operations Menu-------");

Console.WriteLine("1- Reverse Text");

Console.WriteLine("2- Convert to Uppercase");

Console.WriteLine("3- Convert to Lowercase");

Console.WriteLine("4- Go back to previous menu");

Console.WriteLine("=================");

Console.WriteLine("Enter 1,2,3 or 4");

return Console.ReadLine();

}

private static void HandleText()

{

while (true)

{

var choice = HandleTextMenu();// running the sub menu

if (choice == "1" || choice == "2" || choice == "3"|| choice == "4")

{

if (choice == "1")

{

Console.WriteLine("Enter the text you want to reverse:");

string text1 = Console.ReadLine() ?? ""; //Prevent Null, A ?? B, if A empty use B

if (string.IsNullOrWhiteSpace(text1))

{

Console.WriteLine("You did not enter any text. Please try again.");

}

else

{

char[] textR = text1.ToCharArray();

Array.Reverse(textR);

Console.WriteLine("Reversed text: " + new string(textR));

}

}

else if (choice == "2")

{

Console.WriteLine("Enter the text you want to Uppercase");

string text2 = Console.ReadLine()??"";

if (string.IsNullOrWhiteSpace(text2))

{

Console.WriteLine("You did not enter any text. Please try again.");

}

else

{

string textU = text2.ToUpper();

Console.WriteLine("Upper text: " + textU);

}

}

else if (choice == "3")

{

Console.WriteLine("Enter the text you want to Lowercase");

string text3 = Console.ReadLine()??"";

if (string.IsNullOrWhiteSpace(text3))

{

Console.WriteLine("You did not enter any text. Please try again.");

}

else

{

string textL = text3.ToLower();

Console.WriteLine("Lower text: " + textL);

}

}

else if (choice == "4") { Console.Clear(); return; }

}

else

{

Console.WriteLine("Bad input, please try again");

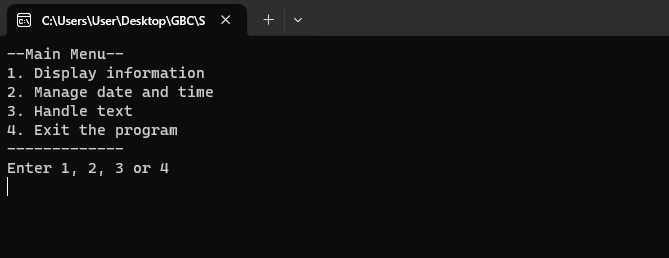
}

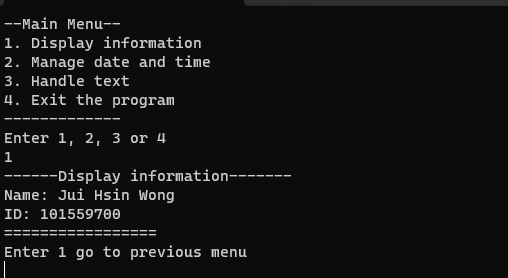
}

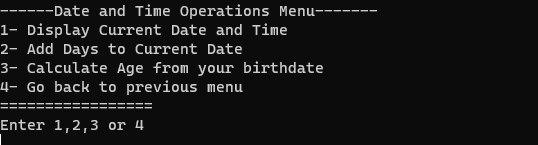
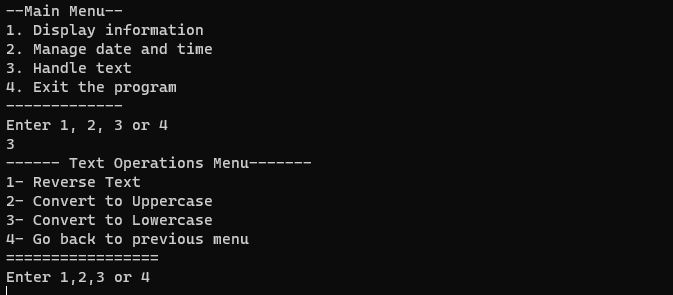
}

}

**Screen shot**

****

****

****

**Assignment: Developing a Menu-Driven Application in C#**

**Objective:**

Develop a C# console application that implements a nested menu system focusing on date and time operations. This application should allow users to navigate through menus and submenus, making selections to perform specific tasks related to date and time management.

**Requirements:**

1. **Menu Structure**: Your application should have a main menu and a submenu under the "Date and Time Operations" option. Each submenu should offer at least three different options or actions.
2. **User Input**: The application must handle user input effectively to navigate through the menus and make selections. Implement input validation and provide clear error messages for invalid inputs.
3. **Functionality**:
   * **Main Menu Options**: Include options to display information, manage date and time operations, handle text operations, and an exit option.
   * **Date and Time Operations Submenu**: This should include options to display the current date and time, add days to the current date, and calculate age from a birthdate.
4. **Defensive Programming and Exception Handling**: Implement defensive programming practices to ensure the program does not crash. Handle all possible exceptions appropriately, ensuring that your application can manage errors gracefully and continue running.
5. **Return Navigation**: Users should be able to return to the previous menu or the main menu from any point within a submenu.
6. **Exit Option**: Include an option to exit the program from the main menu.

**Assignment Deliverables:**

* **Source Code Files**: Well-commented code files explaining the functionality of each part of your menu system.
* **Report**: A document detailing the design of your menu system, challenges encountered during development, the solutions you implemented, and how you ensured the program's stability through defensive programming and exception handling.
* **AI Disclaimer** : If used AI for development of this application you must include the provided form otherwise you may penalized for Academic dishonesty.
* Late submissions will be penalized by 20% on the first day and 10% for each subsequent day.

Example Menu Structure:

Main Menu

|

|-- Option 1: Display Information

|-- Option 2: Date and Time Operations

| |-- Submenu: Date and Time Operations

| |-- Option 1: Display Current Date and Time

| |-- Option 2: Add Days to Current Date

| |-- Option 3: Calculate Age from Birthdate

| |-- Option 4: Return to Main Menu

|-- Option 3: Text Operations

| |-- Submenu: Text Operations

| |-- Option 1: Reverse Text

| |-- Option 2: Convert to Uppercase

| |-- Option 3: Convert to Lowercase

| |-- Option 4: Return to Main Menu

|-- Option 4: Exit