Livia Ferraz Ximenes,

Introduction to Object Programming

420-P25-AS sect. 07056

FINAL PROJECT:

C# FORM APPLICATION DOCUMENTATION

Presented to M. Mihai Maftei

LaSalle College

April 18th, 2016

Contents

[Introduction 3](#_Toc448501914)

[1 Design of the application 4](#_Toc448501915)

[1.1 Money Exchange 5](#_Toc448501916)

[1.2 Temperature Converter 8](#_Toc448501917)

[1.3 Lotto 6-49 12](#_Toc448501918)

[1.4 Simple Calculator 14](#_Toc448501919)

[2 Technical Description 18](#_Toc448501920)

[2.1 Dashboard 18](#_Toc448501921)

[2.1 Dashboard 18](#_Toc448501922)

[2.1 Dashboard 18](#_Toc448501923)

[2.1 Dashboard 18](#_Toc448501924)

[2.1 Dashboard 18](#_Toc448501925)

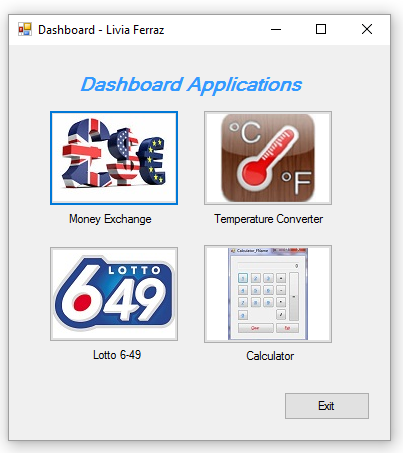
# Introduction

In this document you will find all the pertinent information about the final project from the course Introduction to Object Programming in LaSalle College. This document is divided in two sections. The first section contains print screens of the of the design part of the project showing all functionalities available. The second section depicts the technical aspects of the project, presenting the classes and methods used to build this application.

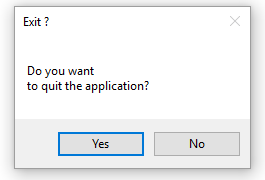
# 1 Design of the application

The application starts with a dashboard that displays the four main functionalities of the program:

* Money exchange converter
* Temperature converter
* Lotto 6-49
* Simple calculator



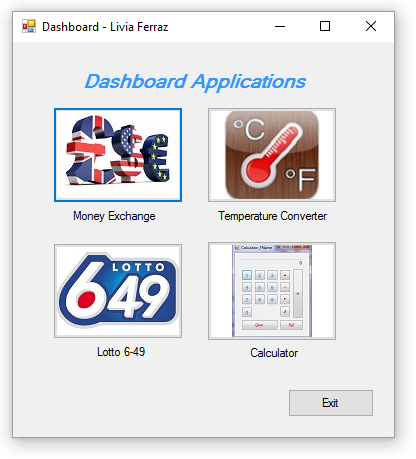
By clicking on the Exit button, a message box is displayed giving the user an option to quit or not the application:

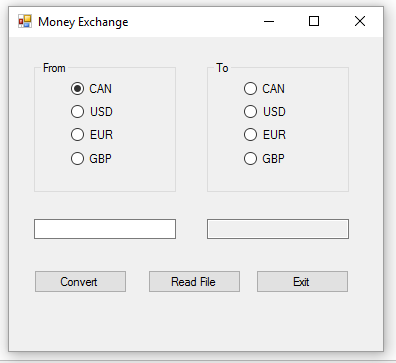


If the user clicks *Yes*, the application closes, and if he clicks *No* the application returns to the dashboard screen.

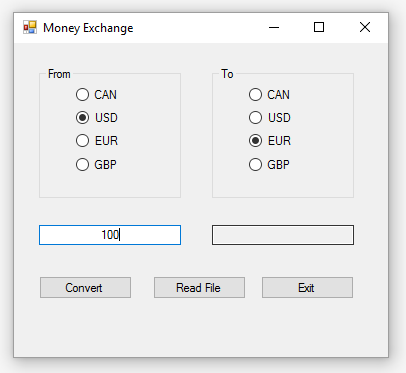
## 1.1 Money Exchange

Clicking on the Money exchange button, a new form opens to the user. This form can be used to make money conversions with four different currencies. The user should choose one currency from the left side group box and one right side group box.

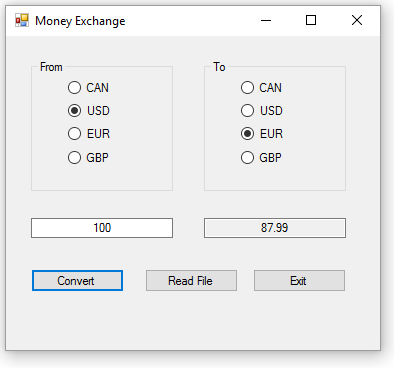




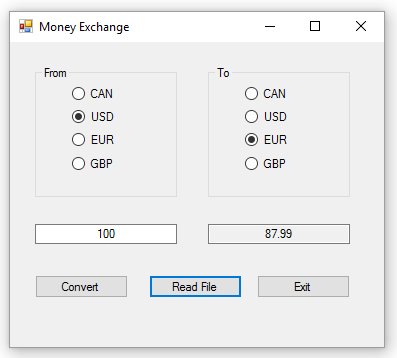
The conversion will be done based on the user’s choice of *From* and *To* currencies. Suppose the user wants to convert U$100 into Euros, he should choose the USD radio button in the left side and EUR radio button in the right side. After it he should type 100 in the text box below and press the button *Convert*.

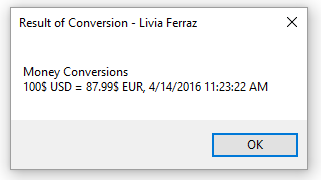


The result of the conversion will appear in the right side text box:



When the button *Read File* is pressed, it opens a message box with the result of all the conversions done and also shows the date and time.

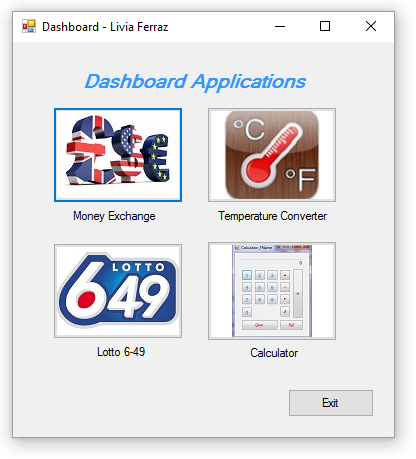




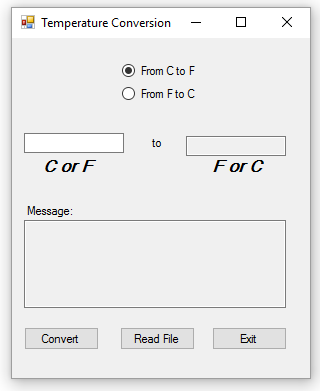
The *Exit* button has the same functionality as the *Exit* button in the Dashboard.

## 1.2 Temperature Converter

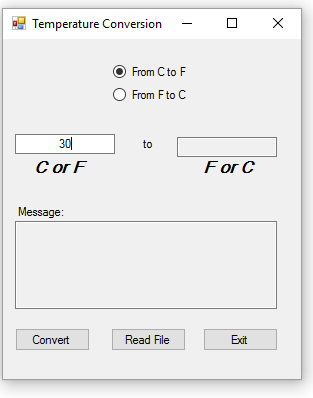
The temperature converter will convert any temperature from Celsius to Fahrenheit and from Fahrenheit to Celsius. To access it, the user has to click the Temperature Converter button in the Dashboard.



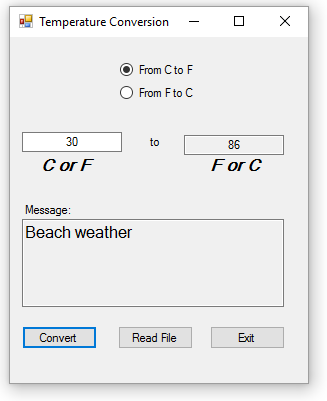
When the application opens, the should choose if he wants to convert from Celsius to Fahrenheit or from Fahrenheit to Celsius by selecting the proper radio button. After it, he should type the temperature he wants to convert in the text box and press the button *Convert.*



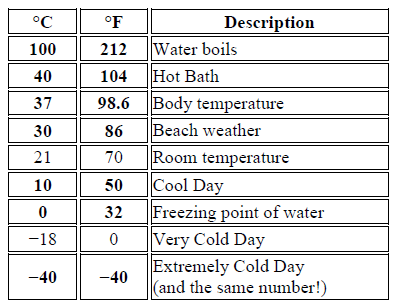
For example, the conversion of 30 degrees Celsius to Fahrenheit would be like this:



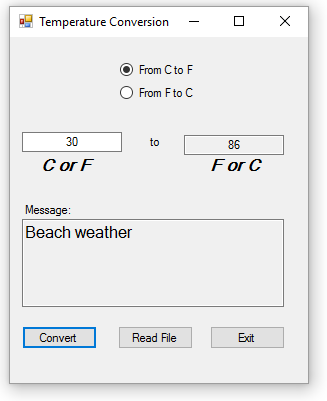
After pressing the *Convert* button, a message is displayed: “Beach Weather”

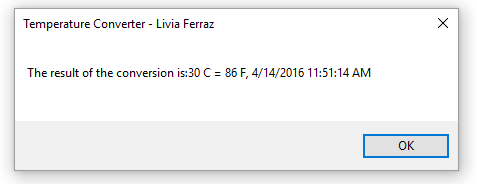


The following messages will appear every time the user converts one of the proposed temperatures:



When the *Read File* button is pressed it will show a message box with the results of all the temperature conversions done by the user.

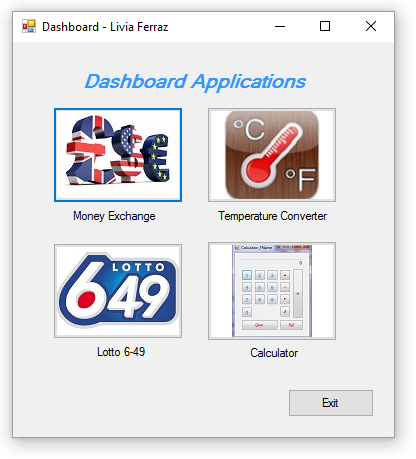




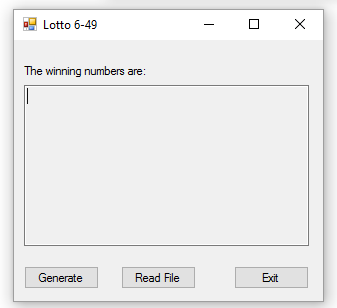
The *Exit* button in the temperature converter has the same functionality as the *Exit* button in the Dashboard.

## 1.3 Lotto 6-49

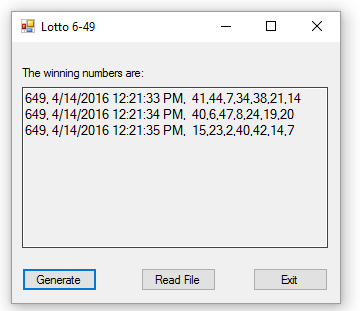
The Lotto 6-49 application simulates a lottery game. It allows the user to generate 7 random numbers for the game Lotto 6/49. To open it, the user should click in the Lotto 6-49 button in the dashboard.



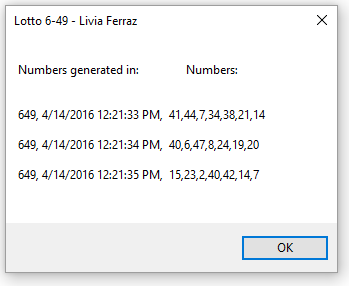
When the application opens, the user should click in the Generate button to generate the lottery numbers.



The user can generate the amount of lottery games he wants. The numbers will be displayed in the text box along with the date and time.



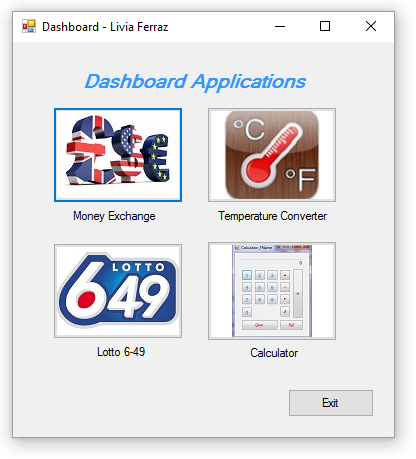
When the *Read File* button is pressed it will show a message box with the results of all the lottery numbers generated by the user.

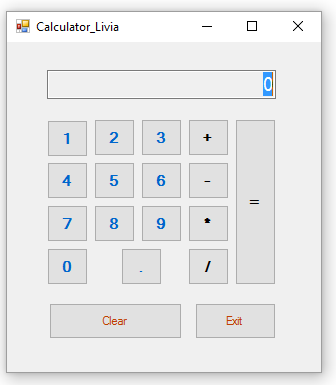


The *Exit* button in the temperature converter has the same functionality as the *Exit* button in the Dashboard.

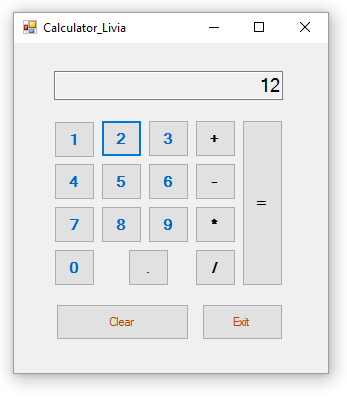
## 1.4 Simple Calculator

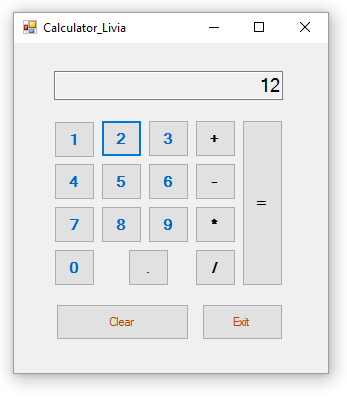
The simple calculator application works like a conventional calculator. The user can access it by clicking in the Calculator button in the Dashboard.

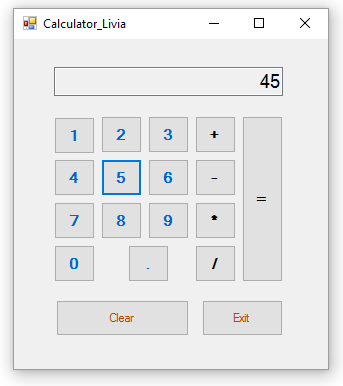
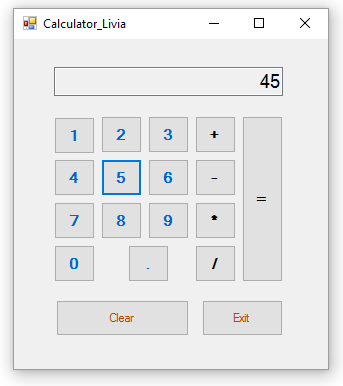




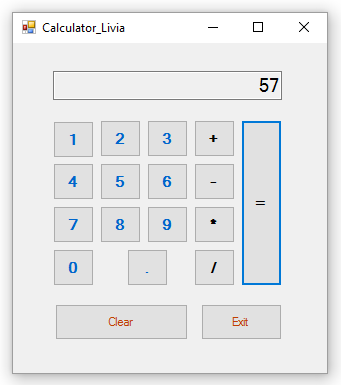
The user should click the number buttons to form the number he wants to calculate. Then he should press the operator. After that the other number and the equal button. For example, if the user wants to add two numbers he has to key in the first number then hit the Add (+) key. After the user enters the final number he should hit the Equals (=) key and the calculator will display the total.



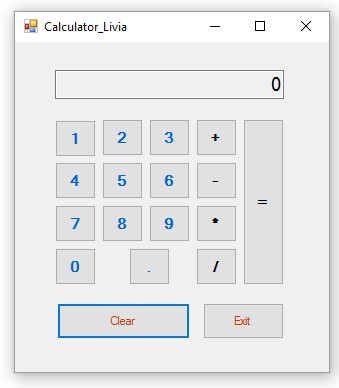




The result (57) is displayed.



The Clear button erases all the values in the calculator.



The *Exit* button in the temperature converter has the same functionality as the *Exit* button in the Dashboard.

# 2 Technical Description

This section of the document contains a description of the classes and methods used to build this application.

## 2.1 Dashboard

|  |  |  |
| --- | --- | --- |
| **Method** | **Code** | **Description** |
| private void button4\_Click | Calculator calc = new Calculator(); | Opens the Simple Calculator application |
| private void button2\_Click | Temperature temp = new Temperature(); | Opens the Temperature converter application |
| private void button1\_Click | Money\_Exchange money = new Money\_Exchange(); | Opens the Money Exchange application |
| private void button3\_Click | Lotto lotto = new Lotto(); | Opens the Lottery application |

## 2.2 Lotto 6-49

|  |  |
| --- | --- |
| **Library** | **Description** |
| using System.IO; | For reading and writing text files |

|  |  |  |
| --- | --- | --- |
| **Class/ Method** | **Code** | **Description** |
| Lotto | public partial class Lotto : Form | Lotto default class |
| Directory | Directory.Exists(dir)  Directory.CreateDirectory(dir); | Creates the directory to save the text files |
| FileStream | fs = new FileStream(path, FileMode.Append, FileAccess.Write); | Creates a new instance of the FileStream class |
| StreamWriter | StreamWriter textOut = new StreamWriter(fs); | Writes characters to a string variable |
| FileStream | fs = new FileStream(path, FileMode.OpenOrCreate, FileAccess.Read); | Reads the generated file |
| StreamReader | StreamReader textIn = new StreamReader(fs); | Reads the generated file |
| List | List<int> randomNumber = new List<int>(); | Creates a list to store the lotto’s numbers |
| Random | Random random = new Random(); | Generates random numbers |
| DateTime | DateTime currentDateTime = DateTime.Now; | Generates the date and time |

|  |  |
| --- | --- |
| **Method** | **Description** |
| private void Lotto\_Load | Creates the directory to save the text files |
| private void button3\_Click | Exit Button |
| private void button1\_Click | Generate button |
| private void button2\_Click | Read file Button |
| this.Close() | Closes the form |

## 2.3 Simple Calculator

|  |  |
| --- | --- |
| **Library** | **Description** |
| using System.IO; | For reading and writing text files |

|  |  |  |
| --- | --- | --- |
| **Class/ Method** | **Code** | **Description** |
| Calculator | public partial class Calculator : Form | Calculator default class |
| Calculator1 | class Calculator1  Calculator1 calc = new Calculator1(); | Stores the variables typed by the user and does the calculations |
| Directory | Directory.Exists(dir)  Directory.CreateDirectory(dir); | Creates the directory to save the text files |
| FileStream | fs = new FileStream(path, FileMode.Append, FileAccess.Write); | Creates a new instance of the FileStream class |
| StreamWriter | StreamWriter textOut = new StreamWriter(fs); | Writes characters to a string variable |

|  |  |
| --- | --- |
| **Method** | **Description** |
| private void Calculator\_Load | Creates the directory to save the text files |
| private void button1\_Click | Number 1 |
| private void button17\_Click | Clear Button |
| private void button18\_Click | Exit Button |
| private void button2\_Click | Number 2 |
| private void button3\_Click | Number 3 |
| private void button8\_Click | Number 4 |
| private void button7\_Click | Number 5 |
| private void button6\_Click | Number 6 |
| private void button12\_Click | Number 7 |
| private void button11\_Click | Number 8 |
| private void button10\_Click | Number 9 |
| private void button16\_Click | Number 0 |
| private void button15\_Click | “ . ” |
| private void AddButton\_Click | + operation |
| private void EqualButton\_Click | Equal |
| private void SubButton\_Click | - operation |
| private void MultButton\_Click | \* operation |
| private void DivButton\_Click | / operation |
| public void Add(decimal displayValue) | Add method in class Calculator1 |
| public void Subtract(decimal displayValue) | Subtract method in class Calculator1 |
| public void Multiply(decimal displayValue) | Multiply method in class Calculator1 |
| public void Divide(decimal displayValue) | Divide method in class Calculator1 |
| public void Clear() | Clear method in class Calculator1 |
| public void Equals(decimal displayValue) | Equals method in class Calculator1, performs all operations |
| .Clear() | Clear the text from the textbox |
| this.Close() | Closes the form |
| EqualButton.PerformClick() | Generates a click event for the equal button |

|  |  |
| --- | --- |
| **Property** | **Description** |
| public decimal CurrentValue | Property for the variable decimal currentValue in class Calculator1 |
| public decimal Operand1 | Property for the variable decimal operand1 in class Calculator1 |
| public decimal Operand2 | Property for the variable decimal operand2 in class Calculator1 |

|  |  |
| --- | --- |
| **Constructor** | **Description** |
| public Calculator1() | Sets the variables to their default values in class Calculator1 |

## 2.4 Money Exchange

|  |  |
| --- | --- |
| **Library** | **Description** |
| using System.IO; | For reading and writing text files |
| using System.Text.RegularExpressions; | To use regular expressions |

|  |  |  |
| --- | --- | --- |
| **Class/ Method** | **Code** | **Description** |
| Money\_Exchange | public partial class Money\_Exchange : Form | Money exchange class |
| Directory | Directory.Exists(dir)  Directory.CreateDirectory(dir); | Creates the directory to save the text files |
| Regex | Regex myRegex1 = new Regex("^([0-9]+)$"); | Performs a validation in the number typed by the user |
| FileStream | fs = new FileStream(path, FileMode.Append, FileAccess.Write); | Creates a new instance of the FileStream class |
| StreamWriter | StreamWriter textOut = new StreamWriter(fs); | Writes characters to a string variable |
| DateTime | DateTime currentDateTime = DateTime.Now; | Shows date and time of the day |
| Convert | can = Convert.ToDouble(textBox1.Text); | Converting to double the variables |
| Math | Math.Round(can, 2).ToString(); | Round numbers to 2 decimals |

|  |  |
| --- | --- |
| **Method** | **Description** |
| private void Money\_Exchange\_Load | Creates the directory to save the text files |
| private void button3\_Click | Exit button |
| private void button1\_Click | Convert Button |
| private void button2\_Click | Read file button |
| private bool ValidNumber | Method to use RegEx |
| private void textBox1\_TextChanged | Performs a validation in the number typed |

## 2.5 Temperature Converter

|  |  |
| --- | --- |
| **Library** | **Description** |
| using System.IO; | For reading and writing text files |

|  |  |  |
| --- | --- | --- |
| **Class/ Method** | **Code** | **Description** |
| Temperature | public partial class Temperature : Form | Temperature converter default class |
| Directory | Directory.Exists(dir)  Directory.CreateDirectory(dir); | Creates the directory to save the text files |
| FileStream | fs = new FileStream(path, FileMode.Append, FileAccess.Write); | Creates a new instance of the FileStream class |
| StreamWriter | StreamWriter textOut = new StreamWriter(fs); | Writes characters to a string variable |
| FileStream | fs = new FileStream(path, FileMode.OpenOrCreate, FileAccess.Read); | Reads the generated file |
| StreamReader | StreamReader textIn = new StreamReader(fs); | Reads the generated file |
| DateTime | DateTime currentDateTime = DateTime.Now; | Generates the date and time |

|  |  |
| --- | --- |
| **Method** | **Description** |
| private void Temperature\_Load | Creates the directory to save the text files |
| private void button3\_Click | Exit Button |
| private void button1\_Click | Convert Button |
| private void button2\_Click | Read file Button |
| this.Close() | Closes the form |
| radioButton1.Checked | Sets a valie when the radiobutton is checked |
| Math.Round | Rounds the answer to two decimals |
| Convert.ToDouble | Converts the variable to a double |