Andrew Hardie 40162946

Coursework 1

SET08119 2017-8 TR1 002 - Object Oriented Software Development

[Year]

# 

Contents

[Student Class 1](#_Toc496772853)

[Form Screenshots 5](#_Toc496772854)

[Main Window Form 5](#_Toc496772855)

[Second Window Form (Show all students) 6](#_Toc496772856)

# Student Class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

//Author name: Andrew Hardie

//Description of class purpose: To set properties for student class objects.

//Date last modified: 25/10/2017

namespace BusinessObjects

{

public class Student

{

//Variables

private int \_matricNo;

private string \_firstName;

private string \_surname;

private int \_courseworkMark;

private int \_examMark;

private string \_dateOfBirth;

private double \_markP;

//Get the students matriculation number and stores the value

public int Matric

{

get

{

return \_matricNo;

}

set

{

//Verifies that the Matric number entered by user is between 10001 - 50000

if (value < 10001 || value > 50000)

{

throw new ArgumentException("Matric number must be between 10001 - 50000" + "\n" + "PLEASE RE-ENTER MATRIC NUMBER" + "\n" + "OR Matric already in use");

}

\_matricNo = value;

}

}

//Get the students firstname and stores the value

public string FName

{

get

{

return \_firstName;

}

set

{

//Validates that the Firstname field isnt blank

if (value == "" || value == "ENTER FIRST NAME...")

{

throw new ArgumentException("Please enter a valid first name");

}

\_firstName = value;

}

}

//Get the students surname and stores the value

public string SName

{

get

{

return \_surname;

}

set

{

//Verifies that the surname field isnt blank

if (value == "" || value == "ENTER SURNAME...")

{

throw new ArgumentException("Please enter a valid surname");

}

\_surname = value;

}

}

//Get the students coursework mark and stores the value

public int CWorkMark

{

get

{

return \_courseworkMark;

}

set

{

//Validates that the courwork mark field value is between 0 - 20

if (value < 0 || value > 20)

{

throw new ArgumentException("Coursework mark must be between 0 and 20" + "\n" + "PLEASE RE-ENTER COURSEWORK MARK");

}

\_courseworkMark = value;

}

}

//Get the students exam mark and stores the value

public int EXMark

{

get

{

return \_examMark;

}

set

{

//Validates the exam mark field by making sure value is between 0 - 40

if (value < 0 || value > 40)

{

throw new ArgumentException("Exam mark must be between 0 and 40" + "\n" + "PLEASE RE-ENTER EXAM MARK");

}

\_examMark = value;

}

}

//Get the students date of birth and stores the value

public string DOB

{

get

{

return \_dateOfBirth;

}

set

{

//Verfies that the DOB field isnt blank

if (value == "")

{

throw new ArgumentException("Please enter a valid date of birth");

}

\_dateOfBirth = value;

}

}

//Doesnt need validation as it takes it value from EXmark % CWorkMark which are already validated, there is no direct user input to this field.

public double GetMark

{

get

{

return \_markP;

}

set

{

\_markP = value;

}

}

//Fetches specific information from student object

public virtual string getDetails()

{

return "Student Record: \n" + "\n" + "Matriculation Number: " + Matric + "\n" + "Firstname: " + FName + "\n" + "Surname: " + SName + "\n" + "Date of Birth: " + DOB + "\n" + "Coursework Mark: " + CWorkMark + "\n" + "Exam Mark: " + EXMark + "\n" + "Overall Grade " + Math.Round(GetMark) + "%";

}

//Fetches specific information from student object

public virtual string getMatricInUse()

{

return "Matriculation Number Already in use: " + Matric + "\n" + "Please use a different matriculation number";

}

//Fetches specific information from student object

public virtual string getAllStudents()

{

return "\n" + "\n" + "Matriculation Number: " + Matric + "\n" + "Firstname: " + FName + "\n" + "Surname: " + SName + "\n" + "Date of Birth: " + DOB + "\n" + "Coursework Mark: " + CWorkMark + "\n" + "Exam Mark: " + EXMark + "\n" + "Overall Grade " + Math.Round(GetMark) + "%";

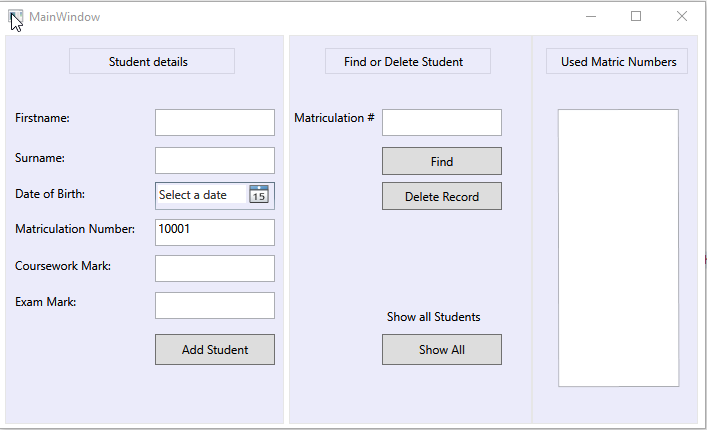
}

}

}

# Form Screenshots

## Main Window Form



## 

## Second Window Form (Show all students)

