ASSIGNMENT 1 FRONT SHEET

Qualification	**BTEC Level 5 HND Diploma in Computing**		
Unit number and title	Unit 1: Programming		
Submission date		**Date Received 1st submission**	
Re-submission Date		**Date Received 2nd submission**	
Student Name	Bùi Hương Linh	**Student ID**	GBH200662
Class	GCH1002	**Assessor name**	Lecturer Mạnh

Student declaration

I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.

Student's signature	

Grading grid

P2	P3	P4	P5	M2	M3	M4	D2	D3	D4

☼Summative Feedback:		☼Resubmission Feedback:
Grade:	**Assessor Signature:**	**Date:**
Lecturer Signature:		

Contents

Task 1: Introduction to your program

I. Introduce the Overview

- Create a student management software program:
  The software includes personal information and scores for three subjects: math, physics, and chemistry. The program must sort and display the list of students on the screen.

II. List out application's requirements

- Fill in student information
- Math scores, physics grades, chemistry grades
- The app can add students, update, delete, and sort.
- The application is built on C sharp computer language
- It is win form application.

Task 2: Explain programming paradigms(P2)

As we are aware of it. The information technology industry has expanded rapidly and widely all over the world. Because of the attractive salary, it is a popular career choice. However, in order to work in this industry, we must have professional skills and qualifications. As a result, anyone working in this industry must be fluent in one or more of the following programming languages: procedural programming, object-oriented programming, and event-driven paradigms,.... These are extremely important programming languages in this field because they assist us in designing a system or a large program. However, each of them has a distinct personality and running speed. To understand them better, I will explain each programming language sand their characteristics.

I. Explain what is Procedural Programming with source code and illustrations

- Procedural programming (POP) is where the major focus on performing tasks in a sequential order. It divides a large program into small functional blocks or functions for ease of programming and testing easier.(Learn Computer Science,2021)

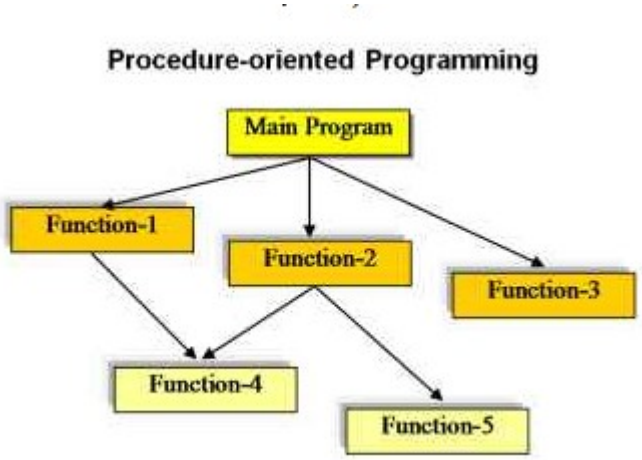


Image 1: Procedure Programming(source: Internet)

- Characteristies:
    - Focus on the work to be done (algorithms).

      + Helps beginners can improve their mindset about solving problems.
    - Large program is divided into subroutines, each of which can be called one or more times in any order.

      +It makes it easier for programmers to address problems since faults in each sub-program may be readily fixed.
    - Most functions use common data.
    - Data in the system is moved from one function to another.

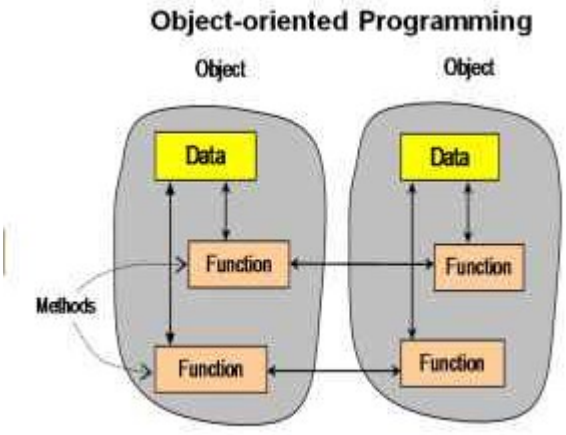
      + Programmers can manage data easily.
    - Uses immutable data.(Leonila Cordrey,2021)

- Example:

```c
#include <stdio.h>

int max (int x,int y,int z){

    int max = x;
    if (y > max){
        max = y;
    }
    if (z > max){
        max = z;
    }
    return max;
}


int main(){
    int x,y,z;
    printf("Enter x: \n");
    scanf("%d", &x);
    printf("Enter y: \n");
    scanf("%d", &y);
    printf("Enter z: \n");
    scanf("%d", &z);

    printf("\n Max = %d\n", max(x,y,z));
}
```

II. Explain what is Object-Oriented Programming with source code and illustrations

- Object-oriented programming (OOP) is the basic and most popular programming paradigm used by most developers. Object-oriented programming is a programming method based on the concept of classes and objects. It focuses on manipulating objects rather than the logic for manipulating them, making the code manageable, reusable, and maintainable.
- This programming approach was developed to reduce some of the drawbacks encountered in the Procedure Oriented Programming Approach.
  (Erin Doherty, 2021)



Imge 2: Object-oriented Programming(source: Internet)

- Characteristies: (Erin Doherty, 2021)

    - Encapsulation: This helps increase the security of the object and avoid the situation of data being unintentionally damaged.

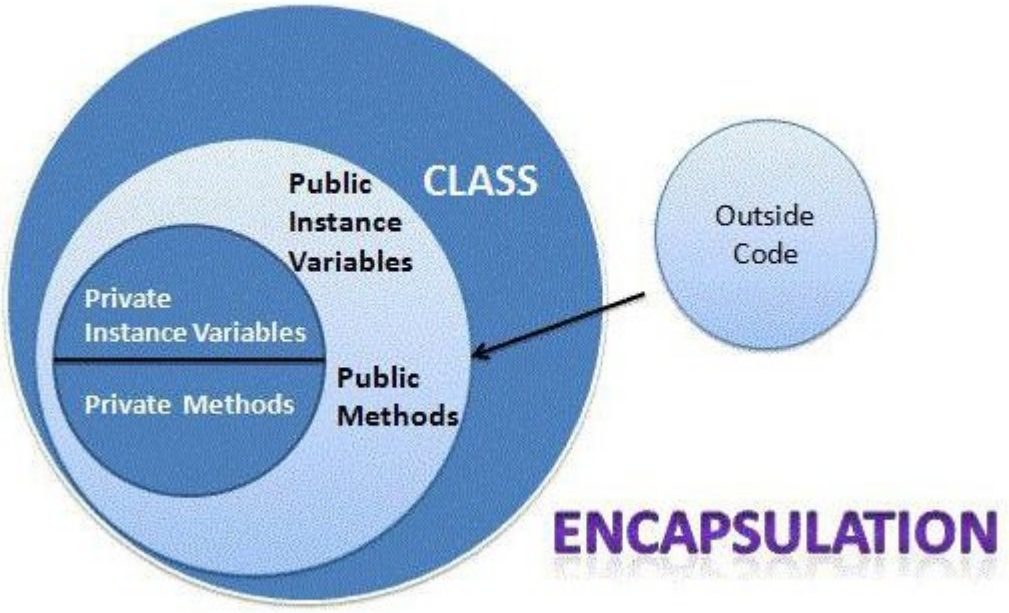


Image 3: Encapsulation (source: Internet)

- Polymorphism: Different objects can perform the same function in different ways.

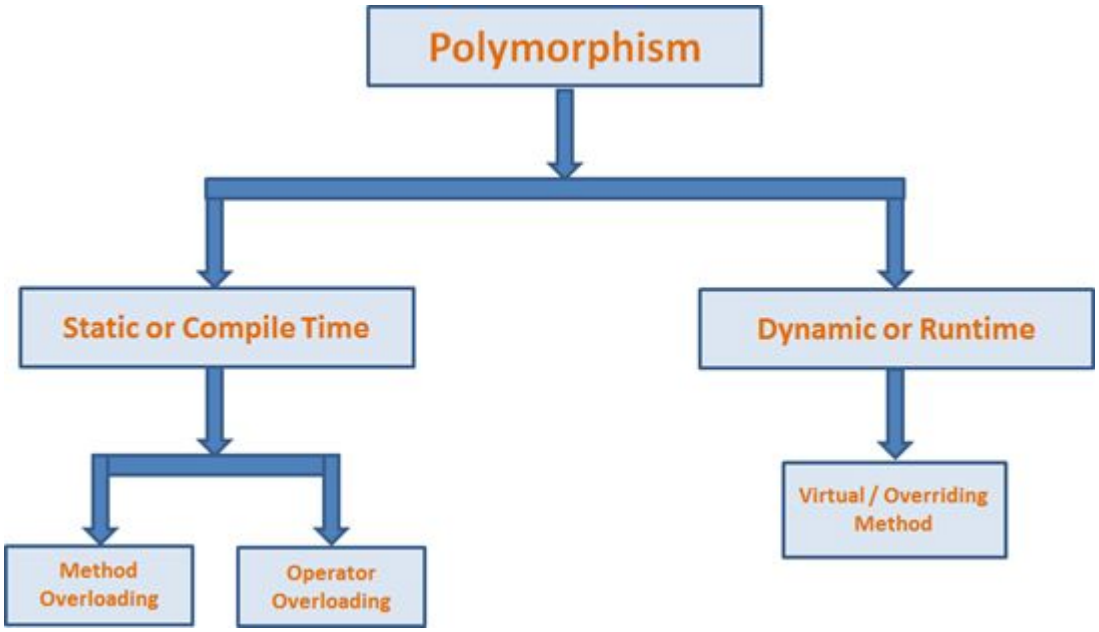


Image 4: Polymorphism (source: Internet)

- Inheritance: This is how a class can inherit properties and methods from another class and use them as its own.

```
class Animal
{
    protected double Weight;
    protected double Height;
    protected static int Legs;

    public void Info()
    {
        Console.WriteLine(" Weight: " + Weight + " Height: " + Height + " Legs: " + Legs);
    }
}
class Cat : Animal
{
    public Cat()
    {
        Weight = 500;
        Height = 20;
        Legs = 2;
    }
}
class Demo
{
    static void Main(string[] args)
    {
        Cat BlackCat = new Cat();

        /* Lớp Cat kế thừa phương thức Info từ lớp Animal nên đối tượng thuộc lớp Cat có thể gọi phương thức Info() */

        BlackCat.Info();
    }
}
```

Image 5: Inheritance

III. Explain what is Event-Driven Programming with source code and illustrations

- Event-driven programming is an important concept in application development and other programming styles, and it generates event handlers and other resources.(Copyright Techopedia, 2021)

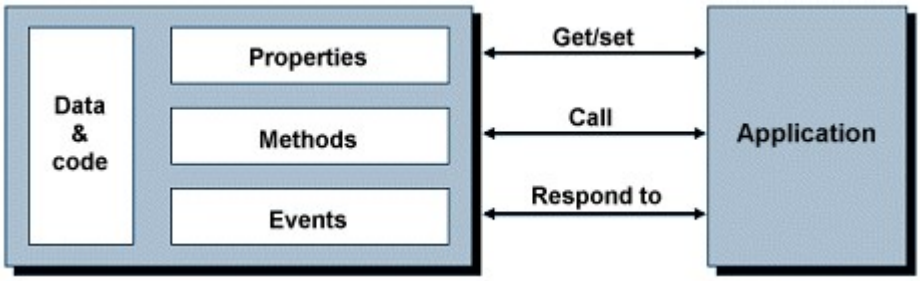


Image 6: Event-Driven Programming(source: Internet)

- Features And Characteristics

- Service-Oriented
  - Service-focused programming is a key feature in event-driven programming, which is used to create programs for services and does not slow down the computer. The service-oriented only consume a small portion of the computer control vitality, and services are typically run in the background of the operating system.
- Events
  - The mouse, computer keyboard, and user interface are examples of events that must be triggered in this program. This means that the user must interact with an object in this program, such as clicking a button with a mouse, selecting a button with the computer keyboard, and so on.
- Trigger functions
  - In event-driven coding, trigger functions are functions that determine what code to execute when a specific event occurs. When a specific event occurs, programmers use these functions to determine which event handler to use for the function.

- Event-Driven Programming vs. Object-Oriented Programming

- Procedural language is a list of instructions telling a computer, step-by-step, what to do, in an order of how to performance the first code to the second and so forth which may contain loops. Procedural programming languages include

C, C++, Fortran, Pascal, these are very detailed and takes a lot of time to write. It relies on the programmer to provide a solution to a problem or to provide an answer. It rotates around keeping code as short as possible and contains a sequence of steps to be carried out. Procedural language is used in the Linux Kernel and Quake III Arena. This can be used to create small projects because it is simple and does not require commands to be entered. Procedural programming does not allow the user to copy and paste the code onto another program also procedural is difficult to understand and takes time to getting used to.

- The approach with OOP, on the other hand, is to discover the elements involved in the problem (people, documents, products, and so on) by analyzing the data (properties) that each one manages, along with the methods (which are nothing other than functions or methods). Then you create derived classes (subclasses) of the main classes (for example, people: employees, suppliers, customers, etc.) that reuse (inherit) the main class's methods and properties and add the specific ones.
- Procedures, objects, methods, and so on "fire" when a "event" (mouse click, keyboard click, window activation, etc.) occurs in event-oriented programming, which is common when programming a GUI. In general, the code is distributed in the application in a structured manner (modules, libraries) where classes and subclasses are defined. If the application must work with events, the events will be associated with the procedures, functions, or methods that manage them in some part of the code.
(Anon, 2021)

IV. Conclude which paradigms will be used to develop the application with explanation

- Object-oriented programming will be used to develop a student management application that I am about to do. Because we will use objects, classes, properties, storage, and computation methods to design and develop software.

Task 3: IDE features(P3-P4-P5)

I. Introduce what is IDE

- The IDE enables programmers to unify the various aspects of writing a computer program. It is also a software application that gathers all of the tools required for a software development project into a single location.(Aaron Walker, 2021)

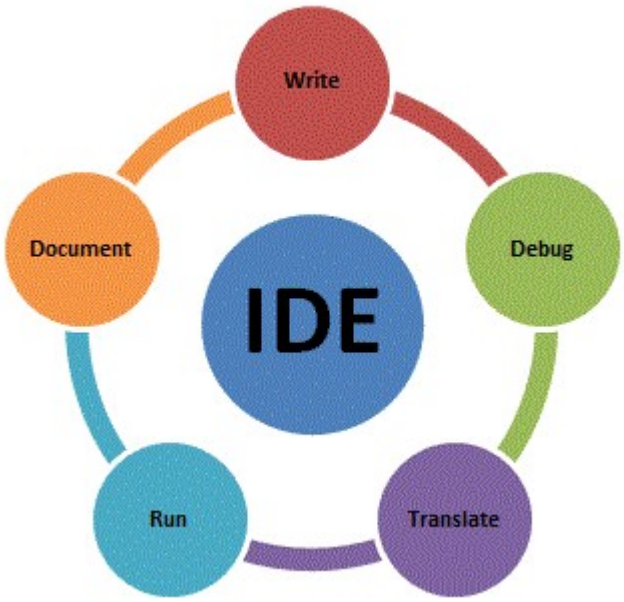


Image 7: IDE(source: Internet)

II. Introduce features of IDE with illustrations (Aaron Walker, 2021)

- Text editor: Virtually every IDE will have a text editor designed to write and manipulate source code. Some tools may have visual components to drag and drop front-end components, but most have a simple interface highlighting language-specific syntax.
- Debugger: Debugging tools assist users in identifying and remedying errors within source code. They often simulate real-world scenarios to test functionality and performance. Programmers and software engineers can usually test the various code segments and identify errors before the application is released.
- Compiler: Compilers are components that translate programming language into a form machines can process, such as binary code. The machine code is analyzed to ensure its accuracy. The compiler then parses and optimizes the code to optimize performance.
- Code completion: Code complete features assist programmers by intelligently identifying and inserting common code components. These features save developers time writing code and reduce the likelihood of typos and bugs.
- Programming language support: IDEs are typically specific to a single programming language, though several also offer multi-language support. As such, the first step is to figure out which languages you will be coding in and narrow your prospective IDE list down accordingly. Examples include Ruby, Python, and Java IDE tools.
- Integrations and plugins: With the name integrated development environment, it is no surprise that integrations need to be considered when looking at IDEs. Your IDE is your development portal, so incorporating all your other

development tools will improve development workflows and productivity. Poor integrations can cause numerous issues and lead to many headaches.

Image 8: Features of IDE(source: Internet)

- Advantage of IDE: (Aaron Walker, 2021)

  + Serves as a single environment for most, if not all, of a developer's needs, such as version control systems, debugging tools, and Platform-as-a-Service (PaaS).

  + Code completion capabilities improve programming workflow.

  + Automatically checks for errors to ensure top-quality code.

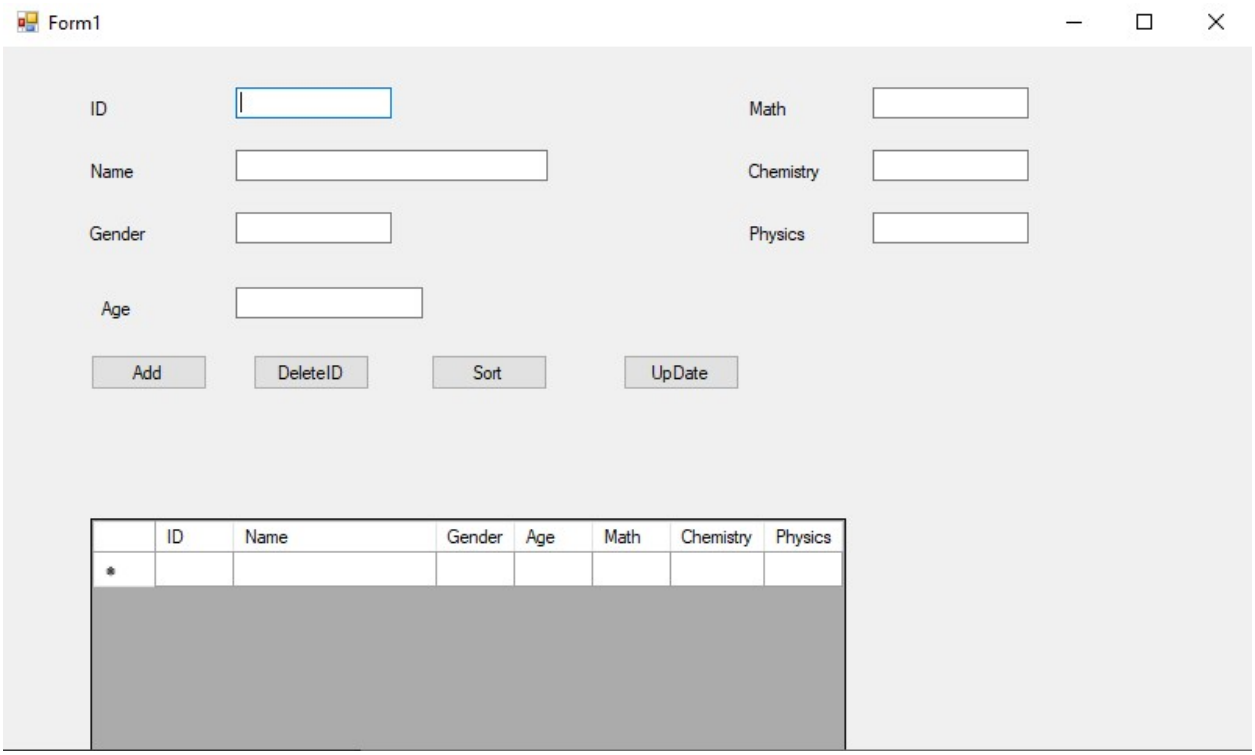
  + Refactoring capabilities allow developers to make comprehensive and mistake-free renaming changes.

  + Maintain a smooth development cycle.

  + Increase developer efficiency and satisfaction.

  + Deliver top-quality software on schedule.

III. Write a program that implements an algorithm using an IDE.(P3)



This is a student management system that I created using a winform interface and visual code. I created the above student attributes on an interface, including id, name,...at a glance, we can also see the text boxes, buttons, and even the datagridview of the list of employees, making it simple to use for everyone. I write in the console and use the list type to use, manage, and save time. I like using the list type because it's fast and easy to use, and I can freely access the number of elements, whereas array is difficult and troublesome, and it has limited elements. Now I'm going to analyze the code so we can figure out how it works.

III.1/ Source code

```
6 references
class SinhVien
{
    4 references
    public int ID { get; set; }
    6 references
    public string Name { get; set; }
    4 references
    public string Gender { get; set; }
    4 references
    public int Age { get; set; }
    4 references
    public double Math { get; set; }
    4 references
    public double Chemistry { get; set; }
    4 references
    public double Physics { get; set; }
    0 references
    public SinhVien()
    {
        ID = 0;
        Name = "No name";
        Gender = "No gender";
        Age = 0;
        Math = 0;
        Chemistry = 0;
        Physics = 0;
    }
    1 reference
    public SinhVien(int svID, string svName, string svGender, int svAge, double svMath, double svChemistry, double svPhysics)
    {
        ID = svID;
        Name = svName;
        Gender = svGender;
        Age = svAge;
        Math = svMath;
        Chemistry = svChemistry;
        Physics = svPhysics;
    }
}
}
```

- Perhaps we are overly familiar with the public and private keywords in OOP; they are keywords that refer to the scope of access; the difference here is that when a function or variable is declared public, other classes can call and use it directly. But private is different in that it is only used locally, which means that other things cannot access it. From here, we can imagine encapsulation, which will be limited in scope. We can understand the set and get functions simply to put the value in or get the value out, it serves for private (encapsulation property), after we put the get, set function in, the access scope will no longer be limited. Here, we can clearly see the similarity that the parameters of the method and the data of the worker have the same name, so the this keyword serves as a distinction between the two and helps to bring more rigor.

```
namespace ASM2
{
    3 references
    public partial class Form1 : Form
    {
        List<SinhVien> lstSinhVien;
        int index;
        1 reference
        public Form1()
        {
            InitializeComponent();
            lstSinhVien = new List<SinhVien>();
        }
```

- Initialize the list of student

III.2/ Checkcontrol

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ASM2
{
    3 references
    public partial class Form1 : Form
    {
        List<SinhVien> lstSinhVien;
        int index;
        1 reference
        public Form1()
        {
            InitializeComponent();
            lstSinhVien = new List<SinhVien>();
        }

        1 reference
        public bool CheckInput()
        {

            if (string.IsNullOrWhiteSpace(txtID.Text))
            {
                MessageBox.Show("You have not entered the Student Id !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtID.Focus();
                return false;
            }
            if (string.IsNullOrWhiteSpace(txtName.Text))
            {
                MessageBox.Show("You have not entered the Student Name !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtName.Focus();
                return false;
            }
            if (string.IsNullOrWhiteSpace(txtGender.Text))
            {
                MessageBox.Show("You have not entered the Student Mark !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtGender.Focus();
                return false;
```

```
            if (string.IsNullOrWhiteSpace(txtAge.Text))
            {
                MessageBox.Show("You have not entered the Student Id !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtAge.Focus();
                return false;
            }
            if (string.IsNullOrWhiteSpace(txtMath.Text))
            {
                MessageBox.Show("You have not entered the Student Name !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtMath.Focus();
                return false;
            }
            if (string.IsNullOrWhiteSpace(txtChemistry.Text))
            {
                MessageBox.Show("You have not entered the Student Mark !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtChemistry.Focus();
                return false;
            }
            if (string.IsNullOrWhiteSpace(txtPhysics.Text))
            {
                MessageBox.Show("You have not entered the Student Mark !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtPhysics.Focus();
                return false;
            }
            return true;
```

- For information, I use the bool data type, which returns true or false.
- The check function allows me to determine whether or not the user has entered sufficient information. If not, the screen will display a message and prompt the user to enter more information. The focus function will me in accomplishing this.

III.3/ Add_Click

```
1 reference
private void btnAdd_Click(object sender, EventArgs e)
{
    if (CheckInput())
    {
        int svID;
        string svName;
        string svGender;
        int svAge;
        double svMath;
        double svChemistry;
        double svPhysics;
        svID = Int32.Parse(txtID.Text);
        svName = txtName.Text;
        svGender = txtGender.Text;
        svAge = Int32.Parse(txtAge.Text);
        svMath = double.Parse(txtMath.Text);
        svChemistry = double.Parse(txtChemistry.Text);
        svPhysics = double.Parse(txtPhysics.Text);
        SinhVien sv = new SinhVien(svID, svName, svGender, svAge, svMath, svChemistry, svPhysics);
        lstSinhVien.Add(sv);
        dataGridView.DataSource = null;
        dataGridView.DataSource = lstSinhVien;
        dataGridView.Refresh();

    }
}
```

- Because txt returns string data, I use the parse keyword to cast string to int and float.
- Then, using the new keyword, I create a new student with the data type student, the variable nv, and all attributes such as id, name, and so on from within the student class.
- Following that, I will use the Add method to add the student to the list and then place them in the datagridview to display the data on the interface.

III.4/ Delete_Click

```
1 reference
private void btnDeleteID_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Do you want to delete?", "Notification", MessageBoxButtons.OKCancel, MessageBoxIcon.Warning) == DialogResult.OK)
    {
        lstSinhVien.RemoveAt(index);
        dataGridView.DataSource = null;
        dataGridView.DataSource = lstSinhVien;

    }
}
```

- Next, I perform a delete operation, and sometimes a user will accidentally press the delete button, so I display a message asking the user if they are certain they want to delete or not.

III.5/ SortByName_Click

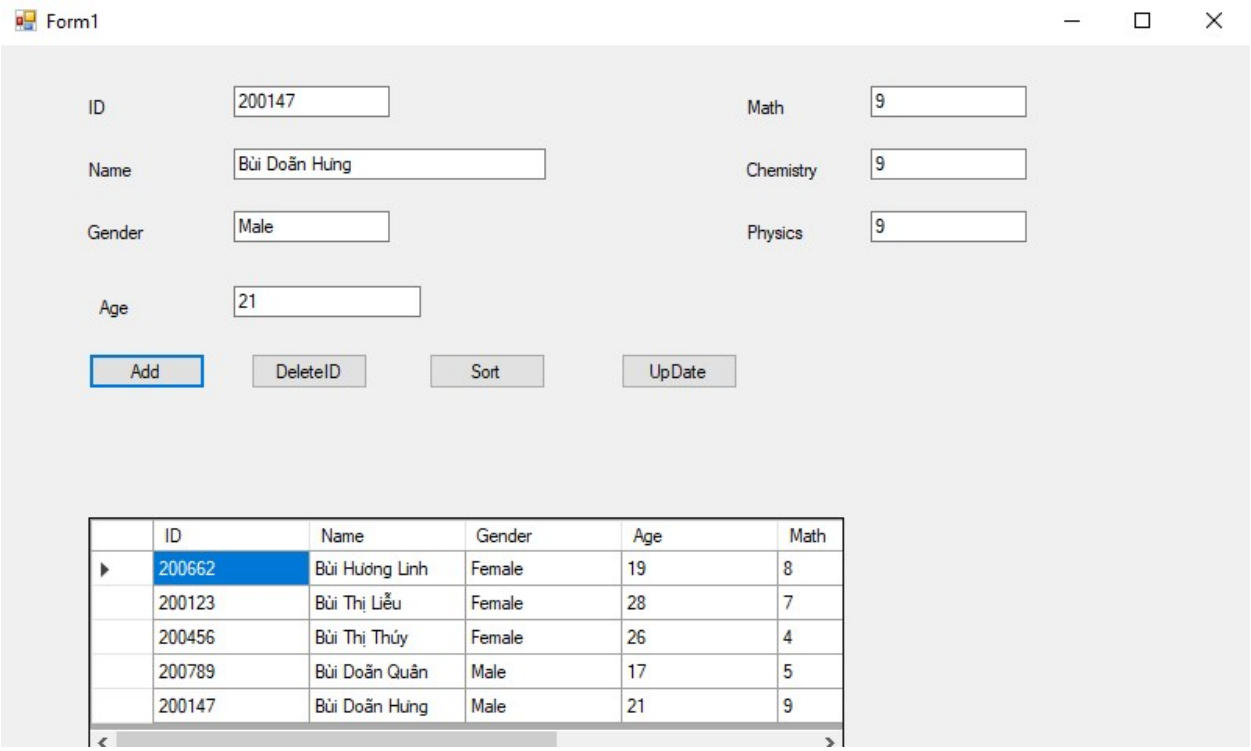
```
126         //sort by name
            1 reference
127         private void btnSortByName_Click(object sender, EventArgs e)
128         {
129             lstSinhVien.Sort((a, b) => a.Name.CompareTo(b.Name));
130             dataGridView.DataSource = null;
131             dataGridView.DataSource = lstSinhVien;
132         }
```

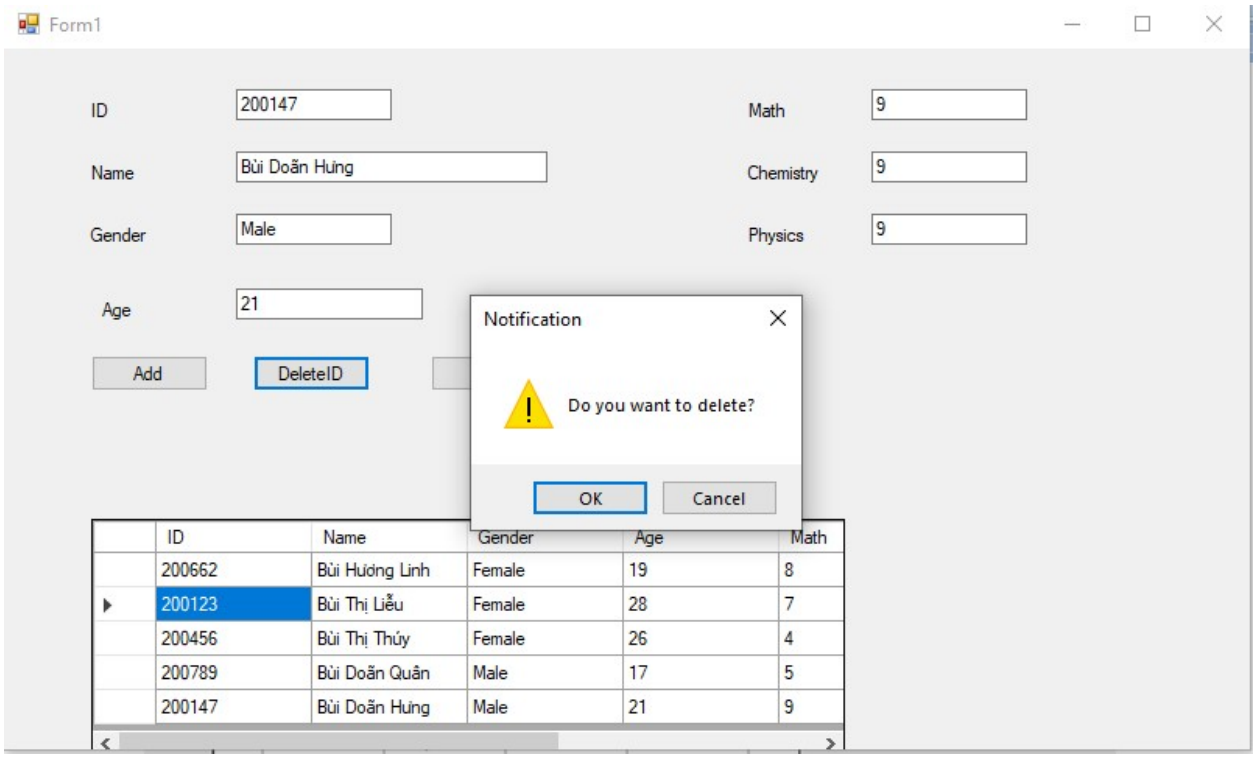
III.6/ dataGridView_CellContentClick:

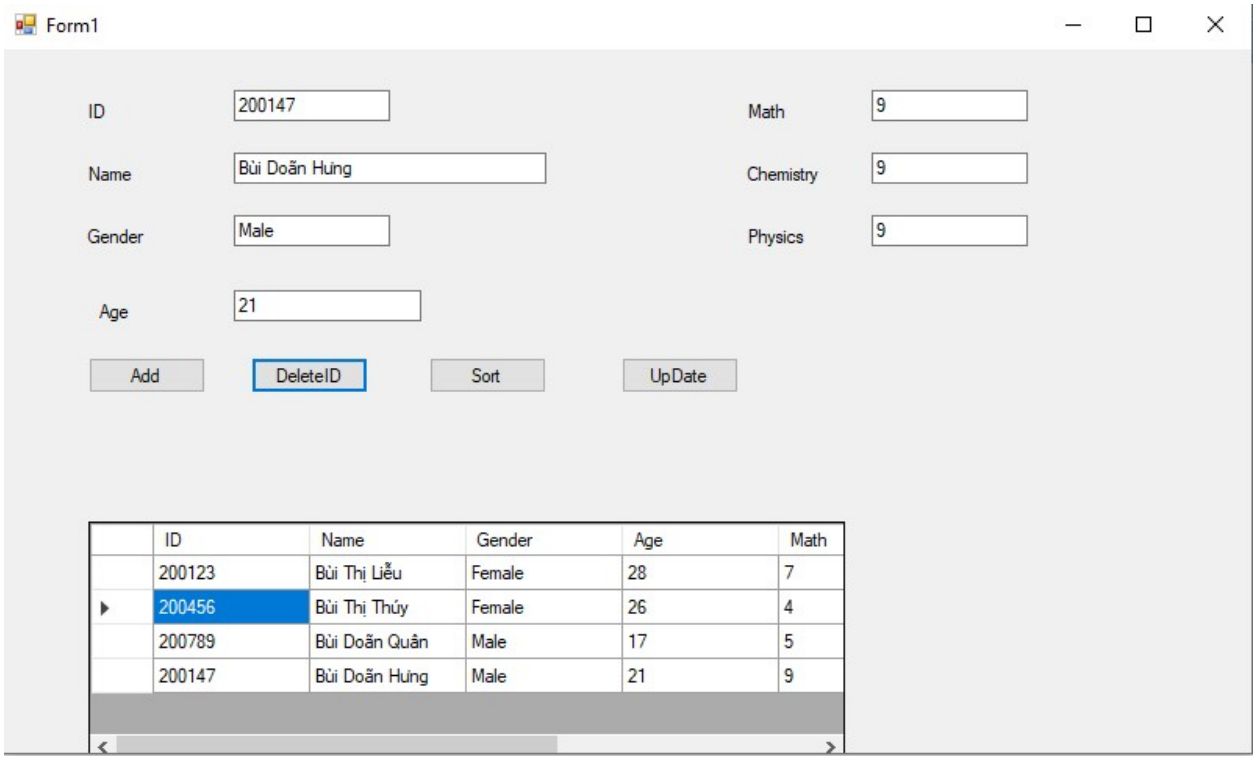
```
1 reference
private void dataGridView_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    index = e.RowIndex;
    if (index >= 0)
    {
        txtID.Text = lstSinhVien[index].ID.ToString();
        txtName.Text = lstSinhVien[index].Name;
        txtGender.Text = lstSinhVien[index].Gender;
        txtAge.Text = lstSinhVien[index].Age.ToString();
        txtMath.Text = lstSinhVien[index].Math.ToString();
        txtChemistry.Text = lstSinhVien[index].Chemistry.ToString();
        txtPhysics.Text = lstSinhVien[index].Physics.ToString();
    }
}
```

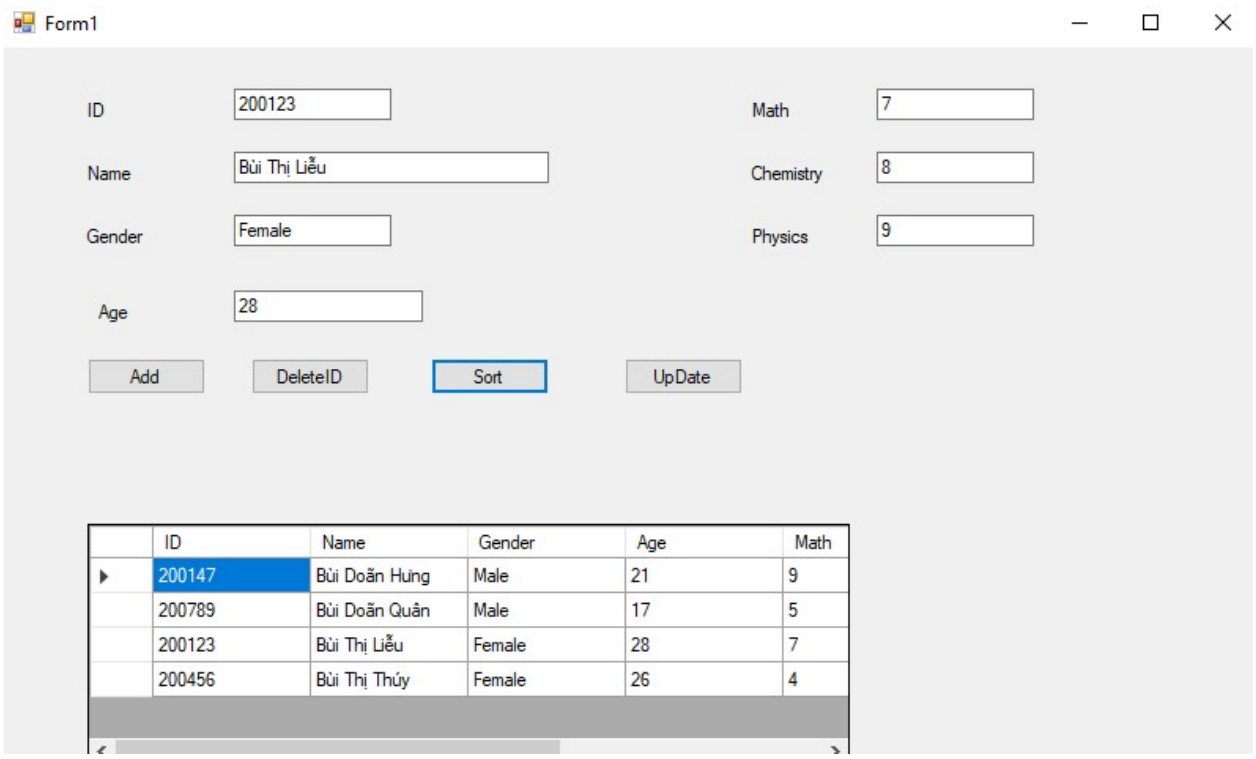
- It supports input data to be filled in to the correct

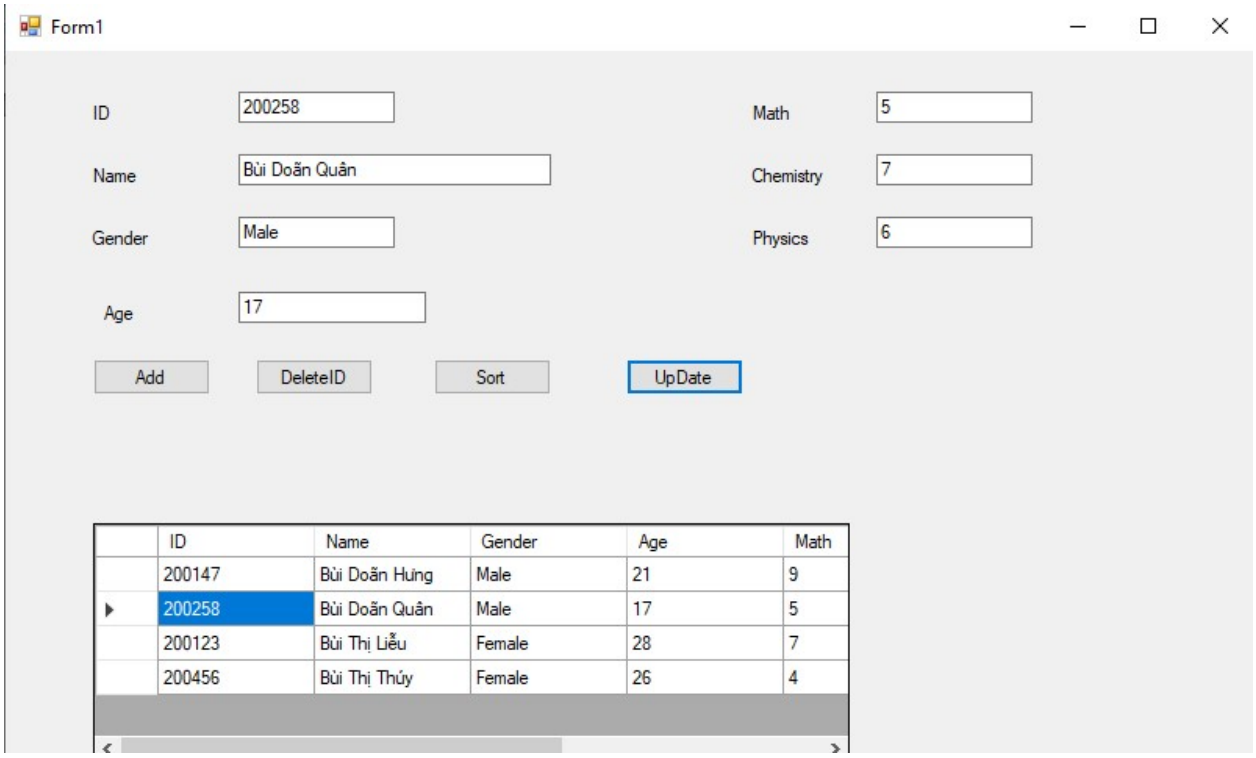
III.7/ Result




Form1
ID
200147
Name
Bùi Doãn Hưng
Gender
Male
Age
21
Math
9
Chemistry
9
Physics
9
Add
DeleteID
Notification
Do you want to delete?
OK
Cancel
ID Name Gender Age Math
200662 Bùi Hương Linh Female 19 8
200123 Bùi Thị Liễu Female 28 7
200456 Bùi Thị Thúy Female 26 4
200789 Bùi Doãn Quân Male 17 5
200147 Bùi Doãn Hưng Male 21 9


Form1
ID
200147
Name
Bùi Doãn Hưng
Gender
Male
Age
21
Math
9
Chemistry
9
Physics
9
Add
DeleteID
Sort
UpDate
ID Name Gender Age Math
200123 Bùi Thị Liễu Female 28 7
200456 Bùi Thị Thúy Female 26 4
200789 Bùi Doãn Quân Male 17 5
200147 Bùi Doãn Hưng Male 21 9


Form1
ID
200123
Name
Bùi Thị Liễu
Gender
Female
Age
28
Math
7
Chemistry
8
Physics
9
Add
DeleteID
Sort
UpDate
ID Name Gender Age Math
200147 Bùi Doãn Hưng Male 21 9
200789 Bùi Doãn Quân Male 17 5
200123 Bùi Thị Liễu Female 28 7
200456 Bùi Thị Thúy Female 26 4


Form1
ID
200258
Name
Bùi Doãn Quân
Gender
Male
Age
17
Math
5
Chemistry
7
Physics
6
Add
DeleteID
Sort
UpDate
ID Name Gender Age Math
200147 Bùi Doãn Hưng Male 21 9
200258 Bùi Doãn Quân Male 17 5
200123 Bùi Thị Liễu Female 28 7
200456 Bùi Thị Thúy Female 26 4

IV. Explain the debugging process and explain the debugging facilities available in the IDE.(P4)

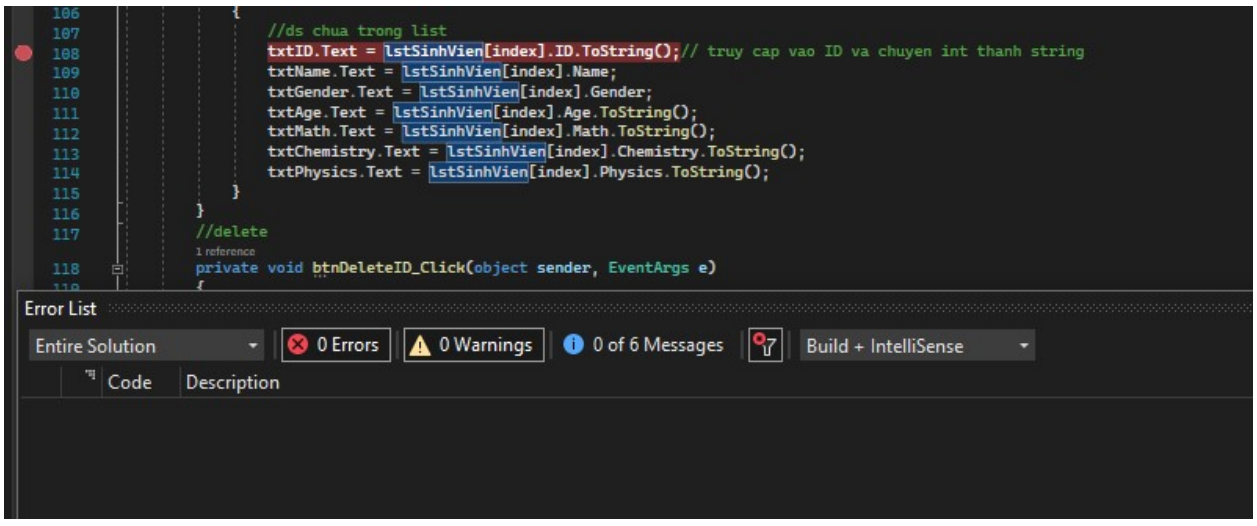
IV.1/ What is the Debug

- Debugging is the process of detecting and eliminating existing and potential errors (also known as "bugs") in software code that can cause it to behave unexpectedly or crash. Debugging is used to find and resolve bugs or defects in software or systems to prevent incorrect operation. Debugging becomes more difficult when various subsystems or modules are tightly coupled, as any change in one module may cause more bugs to appear in another. Debugging a program can sometimes take longer than coding it.(Anon, 2021)

IV.2/ Debugging Process

- Debugging is the process of locating and correcting bugs or errors in any application or software. This process should be completed before releasing software programs or products to the market in order to ensure that they are bug-free.
- Steps:
  - Identifying the error
  - Identifying the error location
  - Analyzing the error
  - Prove the analysis
  - Cover the lateral damage
  - Fix and Validate

IV.3/ Run Debug



```
            //ds chua trong list
            txtID.Text = lstSinhVien[index].ID.ToString();// truy cap vao ID va chuyen int thanh string
            txtName.Text = lstSinhVien[index].Name;
            txtGender.Text = lstSinhVien[index].Gender;
            txtAge.Text = lstSinhVien[index].Age.ToString();
            txtMath.Text = lstSinhVien[index].Math.ToString();
            txtChemistry.Text = lstSinhVien[index].Chemistry.ToString();
            txtPhysics.Text = lstSinhVien[index].Physics.ToString();
        }
    }
    //delete
    1 reference
    private void btnDeleteID_Click(object sender, EventArgs e)
    {
        if (MessageBox.Show("Do you want to delete?", "Notification", MessageBoxButtons.OKCancel, MessageBoxIcon.Warning)
```

```
Output
Show output from: Debug
'ASM2.exe' (CLR v4.0.30319: ASM2.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Con
'ASM2.exe' (CLR v4.0.30319: ASM2.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Cor
'ASM2.exe' (CLR v4.0.30319: ASM2.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml
The program '[13496] ASM2.exe' has exited with code 4294967295 (0xffffffff).
```

V. Outline the coding standard you have used in your code.(P5)

V.1/ what are Coding Standards?

Consider coding standards to be a set of rules, techniques, and best practices for writing cleaner, more readable, and efficient code with fewer errors. They provide a consistent format for software engineers to use when developing sophisticated and highly functional code.

V.2/ Apply coding standard in my code.

- I use pascal case to name the student class and camel case to name the properties aka methods, variables, and so on. It would be more logical to name properties or methods with pascal.

```
class SinhVien
{
    4 references
    public int ID { get; set; }
    6 references
    public string Name { get; set; }
    4 references
    public string Gender { get; set; }
    4 references
    public int Age { get; set; }
    4 references
    public double Math { get; set; }
    4 references
    public double Chemistry { get; set; }
    4 references
    public double Physics { get; set; }
```

- Always align the braces to run the program.

```
1 reference
private void dataGridView_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    index = e.RowIndex;
    if (index >= 0)
    {
        txtID.Text = lstSinhVien[index].ID.ToString();
        txtName.Text = lstSinhVien[index].Name;
        txtGender.Text = lstSinhVien[index].Gender;
        txtAge.Text = lstSinhVien[index].Age.ToString();
        txtMath.Text = lstSinhVien[index].Math.ToString();
        txtChemistry.Text = lstSinhVien[index].Chemistry.ToString();
        txtPhysics.Text = lstSinhVien[index].Physics.ToString();
    }
}
```

- Always separate the methods, different sections of program by one space.

```
1 reference
private void btnDeleteID_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Do you want to delete?", "Notification", MessageBoxButtons.OKCancel, MessageBoxIcon.Warning) == DialogResult.OK)
    {
        lstSinhVien.RemoveAt(index);
        dataGridView.DataSource = null;
        dataGridView.DataSource = lstSinhVien;

    }
}
```

- Making comments allows me to keep track of important information in the code while also making it easier for readers to understand. If I forget or need to refer back to it, I can review it without fear of losing my knowledge.

```
1 reference
private void dataGridView_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    index = e.RowIndex;
    if (index >= 0)
    {
        txtID.Text = lstSinhVien[index].ID.ToString();
        txtName.Text = lstSinhVien[index].Name;
        txtGender.Text = lstSinhVien[index].Gender;
        txtAge.Text = lstSinhVien[index].Age.ToString();
        txtMath.Text = lstSinhVien[index].Math.ToString();
        txtChemistry.Text = lstSinhVien[index].Chemistry.ToString();
        txtPhysics.Text = lstSinhVien[index].Physics.ToString();
    }
}
```

➢ Evaluate Coding Standar:
I feel my code is more standard and smoother now that I've implemented coding standards. I also enjoy writing comments in my code, and I frequently use it when listening to my teacher give a lecture or at the end of a lesson; it not only helps me understand the meaning of the command but also helps me retain knowledge for a longer period of time.

Bibliography

(2021, December 21). Retrieved from Learn Computer Science: https://www.learncomputerscienceonline.com/procedural-programming/?fbclid=IwAR0hkIuXne4jhKyGuapfIGTLkAS4K5bdtplmrmt93AzP5rSAGGaqiVIy6wU

(2021, December 23). Retrieved from veracode.com: https://www.veracode.com/security/integrated-development-environment

(2021, December 26). Retrieved from https://prezi.com/xbpqj5wekzgf/procedural-object-oriented-and-event-driven-programming-paradigms/

Anon. (2021, December 26). Retrieved from elprocus.com: https://www.elprocus.com/what-is-debugging-types-techniques-in-embedded-systems/

Cordrey, L. (2021, December 21). Retrieved from popularask.net: https://popularask.net/which-of-the-following-is-a-main-characteristic-of-procedural-programming-language/?fbclid=IwAR1EOdIZqEnRmfwDA15-V1zZHECGGma92dS13gFxfN8_yiPUP5unXvtaX8M

Doherty, E. (2021, December 21). Retrieved from educative.io: https://www.educative.io/blog/object-oriented-programming?fbclid=IwAR1m0gxx1AGUz5O9ShCBWmfY6WfCKvM2fa_qE6FzGZ4aK_WSSCCkG2bTmls

Techopedia, C. (2021, December 21). Retrieved from techopedia.com: https://www.techopedia.com/definition/7083/event-driven-program?fbclid=IwAR0HS5LQXS3e0mkH5zPizLc2G1d9CoE0_VmKavQD8xpOWXJld_KsYWWB4P8

Walker, A. (2021, December 21). Retrieved from g2.com: https://www.g2.com/articles/ide?fbclid=IwAR0irCNIu44W2MPaxuz0uuFOVmJeXZLY2Fb6AkbFoC-FVytGFkLCR7-UN9U