

MÔN HỌC: LẬP TRÌNH WINDOW 1

BÀI THỰC HÀNH SỐ 7:

LÀM VIỆC VỚI LỚP, ĐỐI TƯỢNG VÀ KHÔNG GIAN TÊN

I. MỤC TIÊU BÀI HỌC

- Kiến thức: Củng cố kiến thức về:
 - Cách thức - cú pháp khai báo lớp - các thành phần của lớp: trường, thuộc tính, phương thức
 - Phương thức getter, setter
 - Tạo đối tượng dẫn xuất từ lớp
- Kỹ năng: Vận dụng tạo - khai báo lớp và thể hiện của lớp
- Thái độ: Tính cẩn thận, chính xác trong công việc

II. NỘI DUNG BÀI HỌC

2.1. Bài tập có hướng dẫn

2.1.1. Ứng dụng Console: Tạo lớp thời gian Time1: Quản lý thời gian: Thiết lập và hiển thị thời gian dạng 12h và 24h

```
class Program
{
    static void Main(string[] args)
    {
        Time1 tg1 = new Time1(14, 12, 12);
        Console.WriteLine("Gio 12 gio: " + tg1.gio12Gio());
        Console.WriteLine("Gio 24 gio: " + tg1.gio24Gio());
        Console.ReadLine();
    }
}

public class Time1 : Object
{
    private int gio;    // 0 -23
    private int phut;   // 0-59
    private int giay;   // 0-59

    public Time1()
    {
        thietlapthoigian(0, 0, 0);
    }

    public int GIO
    {
        get { return gio; }
        set { gio = (value >= 0 && gio < 24) ? value : 0; }
    }

    public int PHUT
    {
        get { return phut; }
        set { phut = (value >= 0 && value < 60) ? value : 0; }
    }

    public int GIAY
    {
        get { return giay; }
        set { giay = (value >= 0 && value < 60) ? value : 0; }
    }
}
```

```

}
public Time1(int gioValue, int phutValue, int giayValue)
{
    this.GIO = gioValue;
    this.PHUT = phutValue;
    this.GIAY = giayValue;
}
public Time1(Time1 t1)
{
    this.GIO = t1.GIO;
    this.PHUT = t1.PHUT;
    this.GIAY = t1.GIAY;
}
// Su dung phuong thuc Setter
public void thietlapthoigian(int gioValue, int phutValue, int giayValue)
{
    GIO = (gioValue >= 0 && gioValue < 24) ? gioValue : 0;
    PHUT = (phutValue >= 0 && phutValue < 60) ? phutValue : 0;
    GIAY = (giayValue >= 0 && giayValue < 60) ? giayValue : 0;
}
//Hien thi gio dang 24 gio
public string gio24Gio()
{
    return String.Format("{0:D2}:{1:D2}:{2:D2}", gio, phut, giay);
}

// Hien thi gio dang 12 gio
public string gio12Gio()
{
    return String.Format("{0}:{1:D2}:{2:D2} {3}",
        ((gio == 12 || gio == 0) ? 12 : gio % 12),
        phut, giay, (gio < 12 ? "AM" : "PM"));
}
}

```

2.1.2. Ứng dụng Windows Form: Tạo lớp thời gian Time1: Quản lý thời gian: Thiết lập và hiển thị thời gian dạng 12h và 24h

Bảng thông tin điều khiển

STT	LOẠI ĐK	TÊN ĐK	TEXT	CHÚ THÍCH
1	Label		GIO - PHUT - GIAY	Font: size: 16
2	Label		Gio	
3	Label		Phut	
4	Label		Giay	
5	Label		Ket qua	
6	Label	lblResult		Borderstyle: fixed single, Autosize: false
7	Button	btnTinh	Tinh	
8	Button	Thoat	Thoat	
9	TextBox	txtGio		
10	TextBox	txtPhut		
11	TextBox	txtGiay		

```

private void btnTinh_Click(object sender, EventArgs e)
{
    Time1 tg1 = new Time1(Int32.Parse(txtGio.Text),
                           Int32.Parse(txtPhut.Text),
                           Int32.Parse(txtGiay.Text));
    lblResult.Text = "Gio 12h: " + tg1.gio12Gio() + "\n" +
                    "Gio 24h: " + tg1.gio24Gio();
}
private void Thoat_Click(object sender, EventArgs e)
{
    this.Close();
}

```

2.1.3. Ứng dụng Console Application: Tạo lớp Diem: Mô tả đặc tính chung cho các đối tượng hình tạo thành từ điểm

```

public class Diem
{
    // Toa do diem
    private int x, y;
    // Constructor
    public Diem() { }
    public Diem(int xValue, int yValue)
    {
        x = xValue;
        y = yValue;
    }
    // Thuộc tính X
    public int X
    {
        get { return x; }
        set { x = value; }
    }
    public int Y
    {

```

```

        get { return y; }
        set { y = value; }
    } // Hiển thị tọa độ điểm
    public override string ToString()
    {
        return "[" + x + ", " + y + "]";
    }
}

```

Tạo lớp Duongtron: kế thừa từ lớp Diem, sử dụng đặc tả tọa độ điểm của lớp Diem, Định nghĩa bổ sung bán kính

// Lớp Duongtron kế thừa từ lớp Diem

```

class Duongtron : Diem
{
    private double bankinh;
    // constructor
    public Duongtron() { }
    public Duongtron(int xValue, int yValue, double bkValue):base(xValue,
yValue)
    {
        BK = bkValue;
    }
    // Thuộc tính
    public double BK
    {
        get { return bankinh; }
        set { if (value >= 0)
                bankinh = value; }
    }
    public double Duongkinh()
    {
        return bankinh * 2;
    }
    public double Chuvi()
    {
        return Math.PI * Duongkinh();
    }
    public double Dientich()
    {
        return Math.PI * Math.Pow(bankinh, 2);
    }
    public override string ToString()
    {
        return "Center = [" + X + ", " + Y + "]" +
            "; Radius = " + bankinh;
    }
}

```

Xử lý sự kiện button click trên Form: Hiển thị thông tin đối tượng dẫn xuất từ lớp Duongtron

```

private void btnDuongtron_Click(object sender, EventArgs e)
{
    // Tạo đối tượng dẫn xuất lớp Duongtron
    Duongtron dtr1 = new Duongtron(1, 1, 2);

    string output = "X: " + dtr1.X + "\n" +
        "Y: " + dtr1.Y + "\n BK: " + dtr1.BK;
}

```

```
output += "Duong kinh " +  
    String.Format("{0:F}", dtr1.Duongkinh()) + "\n";  
output += "Chu vi: " +  
    String.Format("{0:F}", dtr1.Chuvi()) + "\n";  
output += "Dien tích " +  
    String.Format("{0:F}", dtr1.Dientich());  
MessageBox.Show(output, "Minh hoa lop Duong tron");  
}
```

2.2. Bài tập tự làm:

1. Tạo lớp hình chữ nhật kế thừa từ lớp điểm, tính diện tích và chu vi hình chữ nhật
2. Bài toán tạo lớp con người (Person): Trường, thuộc tính và các hàm trong lớp Person
 - Tạo lớp Kiểm tra tạo thể 02 dạng thể hiện của lớp Person
 - Tạo lớp Điểm (gồm tọa độ điểm, phương thức nhập Điểm, hiển thị Điểm ...)
3. Tạo lớp Kiểm tra tạo thể hiện của hiện của Điểm, sử dụng hàm khởi tạo gán giá trị cho Điểm và sử dụng hàm nhập