

BÀI THỰC HÀNH 2

Nội dung: Lớp và đối tượng

- Định nghĩa lớp trong C#
- Sử dụng đối tượng
- Phương thức thiết lập

Bài 2.1. Định nghĩa lớp Point (điểm) gồm có:

- 2 thành phần dữ liệu là tọa độ x và y.
- Các phương thức:
 - o Thiết lập/khởi tạo không tham số gán tọa độ $x=0, y=0$;
 - o Khởi tạo có tham số gán x, y bằng giá trị tham số truyền vào;
 - o In ra màn hình thông tin tọa độ điểm theo định dạng (x,y)
 - o Tính khoảng cách giữa 2 điểm.

Sử dụng lớp Point theo các yêu cầu:

- Tạo một điểm dùng phương thức thiết lập không tham số.
- Tạo một điểm dùng phương thức thiết lập có tham số.
- In ra màn hình thông tin tọa độ 2 điểm và khoảng cách giữa 2 điểm.

Bài 2.2. Định nghĩa lớp Circle theo sơ đồ lớp và mô tả sau:

Circle
- radius: float
+ Circle()
+ Circle(float r)
+ GetRadius(): float
+ SetRadius(float): void
+ GetPerimeter(): double
+ GetArea(): double

- radius bán kính đường tròn
- Circle() phương thức khởi tạo không tham số, gán bán kính bằng 1.
- Circle(float r) phương thức khởi tạo có tham số, gán bán kính bằng r.
- GetRadius() trả về giá trị bán kính đường tròn
- SetRadius(f) gán giá trị bán kính đường tròn bằng một giá trị đưa vào
- GetPerimeter() tính và trả về chu vi đường tròn
- GetArea() tính và trả về diện tích đường tròn

Sử dụng lớp Circle thực hiện các yêu cầu:

- Tạo một đường tròn dùng phương thức thiết lập không tham số.
- Tính và in ra màn hình chu vi, diện tích hình tròn.
- Thay đổi giá trị của bán kính bằng một giá trị nhập vào.
- Tính và in ra màn hình bán kính, chu vi hình tròn.
- Tạo một đường trong bằng phương thức thiết lập có tham số.
- Tính và in ra màn hình chu vi, diện tích hình tròn.

Bài 2.3. Định nghĩa lớp Chuỗi gồm:

- Các thuộc tính: s (chuỗi s), x, y (vị trí xuất hiện của chuỗi s)
- Các phương thức:
 - o Khởi tạo không tham số, khởi tạo có tham số: khởi tạo giá trị cho chuỗi s, vị trí xuất hiện chuỗi tại x, y, màu nền của cửa sổ, màu chữ.
 - o Hiển thị: xóa màn hình, hiển thị chuỗi s ở vị trí x, y; ẩn con trỏ chuột.
 - o Chuyển động: hiển thị chuỗi s ở vị trí x, y với màu chữ, màu nền mặc định trong phương thức khởi tạo; nếu bấm phím ← chuỗi s di chuyển sang trái, bấm phím → di chuyển sang phải, bấm phím ↓ di chuyển xuống dưới, bấm phím ↑ di chuyển lên trên, bấm phím 'Esc' dừng chương trình.

Viết chương trình chính thực hiện:

- Tạo 1 đối tượng chuỗi dùng phương thức thiết lập
- Thực hiện phương thức chuyển động của đối tượng.

Ghi chú: Sử dụng một số thuộc tính, phương thức của Console sau:

- Console.BackgroundColor = ConsoleColor.DarkBlue; // màu nền
- Console.ForegroundColor = ConsoleColor.White; // màu chữ gán là màu trắng
- Console.ForegroundColor = (ConsoleColor)c; //màu chữ là màu c=12
- Console.Clear(); // xóa màn hình
- Console.CursorTop = y; // đưa con trỏ tới dòng y
- Console.CursorLeft = x; // đưa con trỏ tới cột x
- Console.CursorVisible = false; // ẩn con trỏ
- ConsoleKeyInfo phim = Console.ReadKey(true); // lấy thông tin phím bấm
- ConsoleKey.LeftArrow (RightArrow, UpArrow, DownArrow): phím ← (→, ↑, ↓)
- Console.WindowWidth: chiều rộng của cửa sổ
- Console.WindowHeight: chiều cao của cửa sổ

Sử dụng vòng lặp: `while(true){...}`

Bài 2.4. Định nghĩa lớp Adult (người trưởng thành) gồm

- Các thành phần dữ liệu: họ tên, cân nặng, chiều cao.
- Các phương thức:
 - o Khởi tạo không tham số, khởi tạo có tham số, khởi tạo sao chép.
 - o Xuất ra màn hình thông tin người và tình trạng sức khỏe theo chỉ số BMI.

- Tính số cân cần tăng/giảm để người chưa có sức khỏe tốt thành tốt, giả thiết chiều cao không thay đổi.

Sử dụng lớp Adult để thực hiện các yêu cầu:

- Nhập thông tin 1 người (dùng phương thức khởi tạo)
- Xuất thông tin người vừa nhập và tình trạng sức khỏe người đó ra màn hình.
- Nếu người đó chưa có sức khỏe tốt, xuất ra lời khuyên cần tăng/giảm bao nhiêu cân để có sức khỏe tốt.

Ghi chú: Sử dụng một số thuộc tính, phương thức của Console để nhập/xuất tiếng Việt và in ra chữ màu.

- Console.InputEncoding = Encoding.Unicode;
- Console.OutputEncoding = Encoding.Unicode;
- Console.BackgroundColor = ConsoleColor.DarkBlue;// màu nền
- Console.ForegroundColor= ConsoleColor.White;// màu chữ
- Console.Clear();// xóa màn hình

Chỉ số BMI (Body Mass Index) là chỉ số mỡ cơ thể, được tính toán dựa trên trọng lượng tính bằng kilogam và chiều cao tính bằng mét, công thức tính: $BMI = \text{cân nặng} / (\text{chiều cao})^2$. Với

- $BMI < 18.5$: thiếu cân
- $18.5 \leq BMI \leq 25$: bình thường (sức khỏe tốt)
- $25 < BMI < 30$: thừa cân
- $BMI \geq 30$: béo phì

Bài 2.5. Định nghĩa lớp Student (sinh viên) gồm

- Các thuộc tính: id (mã số), name (họ tên), birthday (ngày sinh), male (giới tính nam).
- Các phương thức:
 - Khởi tạo không tham số, khởi tạo có tham số, khởi tạo sao chép.
 - Xuất thông tin sinh viên ra màn hình (mã số, họ tên, giới tính, tuổi)

Sử dụng lớp Student để thực hiện các yêu cầu:

- Nhập thông tin 1 sinh viên (dùng phương thức khởi tạo)
- Xuất thông tin sinh viên vừa nhập ra màn hình.

Ghi chú: Sử dụng một số thuộc tính, phương thức của Console để nhập, xuất tiếng Việt

- Console.InputEncoding = Encoding.Unicode;
- Console.OutputEncoding = Encoding.Unicode;

Bài 2.6. Định nghĩa lớp Time theo sơ đồ lớp và mô tả sau:

Time	
- hour: int - minute: int - second: int	- hour: phần giờ, minute: phút, second: giây - Time() phương thức khởi tạo không tham số, gán hour, minute, second bằng 0. - Time(int h, int m, int s) phương thức khởi tạo có tham số, gán hour bằng h, minute bằng m, second bằng s.
+ Time() + Time(int h, int m, int s) + Hour, Minute, Second + NextSecond(): Time + PreviousSecond(): Time + Show(): void	- Hour, Minute, Second đóng gói các thuộc tính hour, minute, second. - NextSecond() tính và trả về mốc thời gian sau 1 giây. - PreviousSecond() tính và trả về mốc thời gian trước 1 giây. - Show() hiện ra màn hình thời gian theo định dạng h:m:s

Sử dụng lớp Time thực hiện các yêu cầu:

- Tạo một đối tượng thời gian dùng phương thức thiết lập không tham số.
- Tính mốc thời gian sau 1 giây, in đối tượng ra màn hình dùng các thuộc tính đóng gói.
- Tạo một đối tượng thời gian dùng phương thức thiết lập có tham số.
- Tính mốc thời gian trước 1 giây, in đối tượng ra màn hình dùng phương thức Show().

Bài 2.7. Cho định nghĩa lớp PhepToan như dưới đây

```
class PhepToan
{
    //khai báo các thành phần dữ liệu
    float a, b;
    char c;
    //phương thức thiết lập có tham số có giá trị mặc định
    public PhepToan (float a1=0, float b1=0, char c1='+')
    {
        a = a1; b = b1;
        c = c1;
    }
    public void CapNhat()
    {
        Console.Write("nhap 2 so thuc:");
        a = float.Parse(Console.ReadLine());
        b = float.Parse(Console.ReadLine());
        Console.Write("nhap phep toan (+,-,*,/):");
        c = char.Parse(Console.ReadLine());
    }

    //tính toán và trả về giá trị
    float TinhToan()
    {
        float kq = 0;
        switch (c)
        {
            case '+': kq = a + b; break;
            case '-': kq = a - b; break;
        }
    }
}
```

```

        case '*': kq = a * b; break;
        case '/': kq = a / b; break;
    }
    return kq;
}
//xuất ra màn hình kết quả
public void Xuat()
{
    Console.WriteLine($"{a}{c}{b}={TinhToan()}");
}
}

```

Hãy cho biết kết quả chạy các đoạn lệnh sau:

- a) PhepToan pt = new PhepToan();
 pt.Xuat();
- b) PhepToan pt = new PhepToan();
 pt.CapNhat();//giả sử nhập a=7, b=8, c=+
 pt.Xuat();
- c) PhepToan pt = new PhepToan(3,6,'*');
 pt1.Xuat();
- d) PhepToan pt = new PhepToan(3,6,'*');
 pt.CapNhat();//giả sử nhập a=7, b=8, c=+
 pt.Xuat();