

The cover slide features the HUFLIT logo at the top left, followed by the text 'KHOA CÔNG NGHỆ THÔNG TIN'. The main title 'Bài 8' is at the top, followed by 'OOP - MỘT SỐ VẤN ĐỀ KHÁC'. A small note 'Biên soạn: Huỳnh Thành Lộc' is on the right, and the date 'Cập nhật: Tháng 01/2022' is at the bottom left.

1

---

---

---

---

---

---

The slide shows the HUFLIT logo and the title 'Nội dung (01 buổi = 03 tiết)'. Below it is a bulleted list: 'Từ khóa "abstract"', 'Từ khóa "static"', and 'Bài tập vận dụng'. The number '2' is at the bottom right.

2

---

---

---

---

---

---

The slide has a large number '1' in the top left corner. Below it is the text 'TỪ KHÓA "abstract"'. The number '3' is at the bottom right.

3

---

---

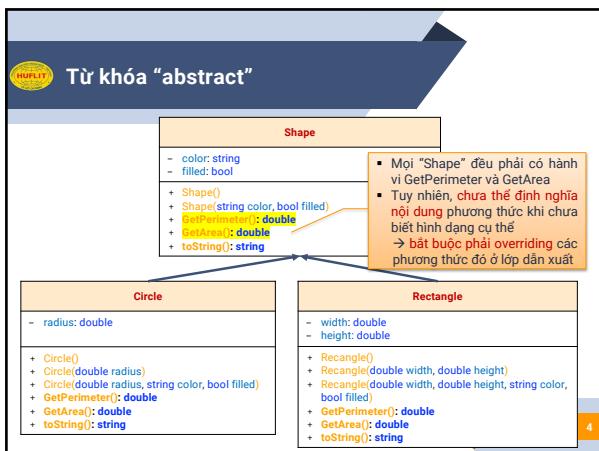
---

---

---

---

# 1250074 - Kỹ thuật lập trình



4

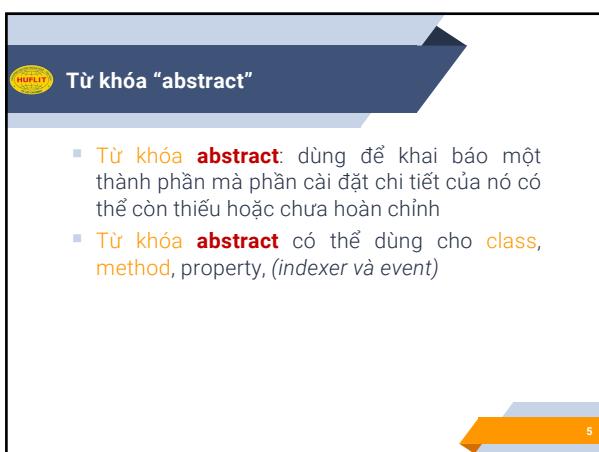
---

---

---

---

---



5

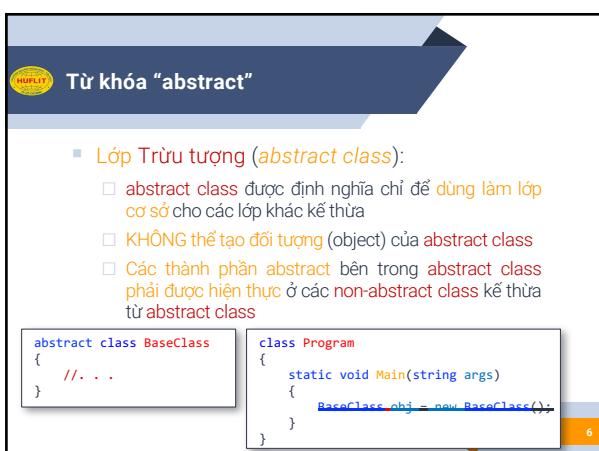
---

---

---

---

---



6

---

---

---

---

---

# 1250074 - Kỹ thuật lập trình

**Từ khóa "abstract"**

- Phương thức Tríu tượng (*abstract method*):
  - abstract method là phương thức chỉ có phần khai báo (prototype) mà không có nội dung → khai báo method kết thúc bằng dấu ";" và không có cặp dấu "{}"
  - abstract method phải được khai báo bên trong abstract class

```
abstract class BaseClass
{
    abstract void MyMethod();
}
```

abstract method bắt buộc phải được hiện thực (overriding) ở các non-abstract class kế thừa từ abstract class

7

---

---

---

---

---

---

**Từ khóa "abstract"**

```
public abstract class Shape
{
    // Attributes
    private string color;
    private bool filled;

    // Constructors
    public Shape() {}
    public Shape(string color, bool filled)
    {
        this.color = color;
        this.filled = filled;
    }

    // Methods
    public abstract double getArea();
    public abstract double getPerimeter();
    public virtual void toString() { . . . }
}
```

8

---

---

---

---

---

---

**Từ khóa "abstract"**

```
public class Circle : Shape
{
    // Attributes
    private double radius;

    // Constructors
    public Circle(double radius)
    {
        this.radius = radius;
    }
    public Circle(double radius, string color, bool filled) : base(color, filled)
    {
        this.radius = radius;
    }

    // Methods
    public override double getArea() { return 3.14 * 2 * radius; }
    public override double getPerimeter() { return 3.14 * radius * radius; }
    public override void toString() { . . . }
}
```

9

---

---

---

---

---

---

1250074 - Kỹ thuật lập trình

## Từ khóa “abstract”

10

# 2

## TỪ KHÓA “static”

11

## Từ khóa “static”

12

# 1250074 - Kỹ thuật lập trình

**Từ khóa "static"**

- static attribute:  

```
public class Staff
{
    public static double baseSalary;
    public double coefficient;
    ...
    public double Pay()
    {
        return baseSalary * coefficient;
    }
}
```

```
public class Program
{
    static void Main(string args)
    {
        Staff.baseSalary = 4000000;
        Staff obj = new Staff();
        obj.coefficient = 3.2;
        obj.Pay();
    }
}
```

  - baseSalary tồn tại không phụ thuộc vào các object của class Staff
  - Truy cập static attribute thông qua class name

13

---

---

---

---

---

13

**Từ khóa "static"**

- static method:  

```
public class Fraction
{
    private int numerator;
    private int denominator;
    ...
    public static Fraction Add(Fraction f1, Fraction f2)
    {
        ...
    }
}
```

```
public class Program
{
    static void Main(string args)
    {
        Fraction f1 = new Fraction();
        Fraction f2 = new Fraction();
        Fraction sum = Fraction.Add(f1, f2);
    }
}
```

  - Truy cập static method thông qua class name
  - Cần phải truyền đủ 2 đối tượng Fraction để thực hiện phép cộng

14

---

---

---

---

---

14

**Từ khóa "static"**

- static class: tất cả thành phần của class đều static → thường dùng để tạo các lớp tiện ích (không cần tạo đối tượng của lớp đó)  

```
public static class Math
{
    public static double Pow(double x, double y)
    {
        return ... ;
    }
    public static double Abs(double value)
    {
        return ... ;
    }
    ...
}
```

```
public class Program
{
    static void Main(string args)
    {
        double n = Math.Pow(2, 3);
        double x = Math.Abs(-5);
    }
}
```

15

---

---

---

---

---

15



16

---

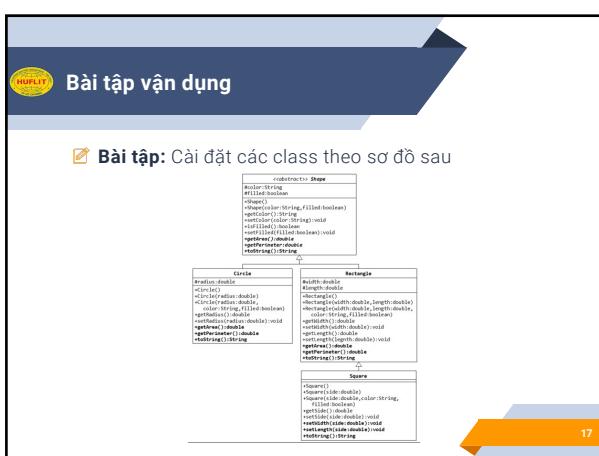
---

---

---

---

---



17

---

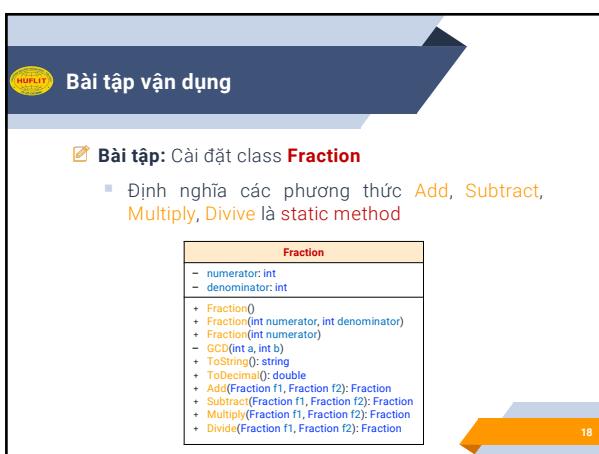
---

---

---

---

---



18

---

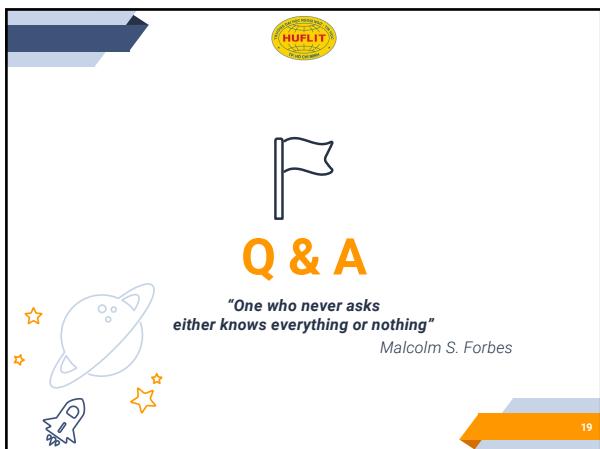
---

---

---

---

---



19

---

---

---

---

---

---

---