

NGÔN NGỮ LẬP TRÌNH C#

CHƯƠNG 2: CÁC LỆNH CƠ BẢN TRÊN C#

Giảng viên biên soạn: ThS Nguyễn Bích Vân

1

NỘI DUNG

- 1. Cấu trúc chương trình C#
- 2. Kiểu dữ liệu, từ khoá, định danh, biến, hằng
- 3. Các toán tử
- 4. Nhập / Xuất Console
- 5. Mảng 1 chiều, 2 chiều
- 6. Kiểu enumeration
- 7. Các thao tác trên chuỗi String
- 8. Cấu trúc điều khiển
- 9. Phương thức và tham số

2

1. Cấu trúc chương trình C#

- File Program.cs là file mặc định chứa hàm Main của chương trình nên chương trình sẽ thực thi các lệnh viết trong hàm Main của file Program.cs.

```
// Khai báo thư viện sử dụng
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1 //tên project
{
    class Program //tên lớp (tên file cs = tên lớp class)
    {
        static void Main(string[] args)
        {
            //Chương trình chính viết trong hàm Main
            Console.WriteLine("HELLO UIT");
        }
    }
}
```

3

Comments (ghi chú)

Kiểu Comments	Ý nghĩa	Ví dụ
Delimited comments	Chú thích có thể mở rộng cho nhiều dòng mã nguồn	<code>/*comment*/</code>
Single-line comments	Chú thích sử dụng cho một dòng mã nguồn	<code>//comment</code>
XML delimited comments	Chú thích sử dụng cho nhiều dòng mã nguồn chuẩn của XML	<code>/**comment**/</code>
XML single-line comments	Chú thích sử dụng cho một dòng mã nguồn chuẩn của XML	<code>///comment</code>

4

2. Kiểu dữ liệu

- Có 2 loại kiểu dữ liệu định sẵn và kiểu dữ liệu tự định nghĩa.
- Trong mỗi kiểu dữ liệu, được phân thành 2 loại: kiểu dữ liệu giá trị (value) và kiểu dữ liệu tham chiếu (reference)
- Kiểu dữ liệu định sẵn gồm các kiểu dữ liệu như: số nguyên (**int**), số thực (**float**), logic (**bool**), ký tự (**char**), chuỗi (**string**), mảng (mảng 1 chiều, 2 chiều), liệt kê (**enum**), cấu trúc (**struct**)

5

2. Kiểu dữ liệu

Kiểu dữ liệu	Ví dụ minh họa
Số nguyên	int x=3;
Số thực	float y=1.2;
Logic	bool z=true;
Ký tự	char c='A';
Chuỗi	string s="Đại học";
Mảng 1 chiều	int [] arr = new int [10];
Mảng 2 chiều	int [,] mt = new int[3, 5];
Liệt kê	enum Ngay {Hai, Ba, Tu, Nam, Sau, Bay, ChuNhat};
Cấu trúc	struct SP { public string ID; public string Name; }

6

2. Từ khóa - KeyWords

- Danh sách các từ khóa:

abstract	add*	as	base	bool	break	byte
case	catch	char	checked	class	const	continue
decimal	default	delegate	do	double	else	enum
event	explicit	extern	false	finally	fixed	float
for	foreach	get*	goto	if	implicit	in
int	interface	internal	is	lock	long	namespace
new	null	object	operator	out	override	params
partial*	private	protected	public	readonly	ref	remove
return	sbyte	sealed	set*	short	sizeof	stackalloc
static	string	struct	switch	this	throw	true
try	typeof	uint	ulong	unchecked	unsafe	ushort
using	value*	virtual	void	volatile	where*	while
yield						

7

2. Định danh - Identifier

- Định danh là việc xác định tên: biến, hàm, hằng, ...
- Phân biệt chữ hoa thường

Quy ước đặt tên:

- Sử dụng 26 chữ cái (thường/ hoa), 10 chữ số
- Dấu nối (_)
- Không dùng chữ số ở đầu
- Không trùng với từ khóa

8

2. Biến và hằng

- Biến dùng để lưu trữ dữ liệu. Mỗi biến thuộc về một KDL nào đó. Cú pháp: **<KDL> tên biến;**
- Ví dụ: `int x = 22;`
 `float y = 1.8;`
- Hằng cũng là một biến nhưng giá trị của hằng không thay đổi. Cú pháp: **<const> <type> <tên hằng> = <giá trị>;**
- Ví dụ: `const int DoSoi = 100;`

9

3. Các toán tử

- Toán tử gán: =
- Toán tử toán học: +, -, *, /, %,
 - Toán tử tăng ++, giảm -- (x++, ++x, x--, --x)
- Tính toán và gán trở lại: +=, -=, *=, /=, %=
- Toán tử quan hệ: ==, !=, >, >=, <, <=
- Toán tử logic: &&, ||, !
- Toán tử 3 ngôi:
 - <Biểu thức điều kiện> ? <Biểu thức thứ 1> : <Biểu thức thứ 2>

10

4. Xuất – Console.System

- Xuất xong không xuống hàng: **Write**
- Xuất xong xuống hàng: **WriteLine**
- Ví dụ 1: Xuất không định dạng

```
double x = 7.534;
string s = "ABC";
Console.WriteLine("x = "+x+"; s = "+s);
```

- Ví dụ 2: Xuất có định dạng

```
float x = 7.53489F;
double y = 5.6482;
Console.WriteLine("x = {0: 0.0000}; y = {1: 0.00} ", x, y);
```

11

4. Xuất – Console.System

- Thiết lập vị trí xuất

```
Console.SetCursorPosition(cột, dòng);
```

- Ví dụ:

Xuất chuỗi Hello vị trí dòng 20 và cột 10

```
Console.SetCursorPosition(10, 20);
```

```
Console.Write("Hello");
```

Ký tự	Ý nghĩa
\'	Dấu nháy đơn
\"	Dấu nháy đôi
\\	Dấu chéo ngược “\”
\0	Null
\a	Alert : Tiếng bíp
\b	Lùi về trước
\f	Form feed
\n	Xuống dòng
\r	Về đầu dòng
\t	Tab ngang

12

4. Nhập – Console.System

- **Mẫu chung:**

<KDL> Biến;

Biến = <KDL>.Parse(Console.ReadLine());

- **Hoặc**

<KDL> Biến;

Biến = Convert.To<KDL.Net>(Console.ReadLine());

```
string s;
int n;
s = Console.ReadLine();
n = int.Parse(s);
```

```
int n;
n = int.Parse(Console.ReadLine());
```

13

5. Mảng

Cấu trúc cú pháp của mảng 1 chiều:

Datatype [] VariableName = new Datatype [Number of elements];

Trong đó:

- Datatype: kiểu dữ liệu mà mảng lưu trữ
- VariableName: tên biến mảng.
- Number of elements: là số phần tử tối đa của mảng

Ví dụ:

int[] arr= new int[5]; → Mảng arr có tối đa 5 phần tử từ 0 đến 4

int[] value= {75, 26, 81, 43, 19}; → value[0] = 75

14

5. Mảng

Cấu trúc cú pháp của mảng 2 chiều:

Datatype [,] VariableName = new Datatype [Rows, Columns];

Trong đó:

- Datatype: kiểu dữ liệu mà mảng lưu trữ
- VariableName: tên biến mảng.
- Rows: là số lượng dòng
- Columns: là số lượng cột

Ví dụ:

`int[,] arr = new int[5,3];` → Mảng arr gồm 5 dòng 3 cột

15

6. Kiểu liệt kê enumeration

- Enum: Kiểu liệt kê đơn giản là tập hợp các tên hằng có giá trị không thay đổi (thường được gọi là danh sách liệt kê).

- `const int DoNong = 60;`
- `const int DoAm = 40;`
- `const int DoNguoi = 20;`



```
enum NhietDo : int
{
    DoDong = 0,
    DoNguoi = 20,
    DoAm = 40,
    DoNong = 60,
    DoSoi = 100,
}
```

16

7. Thao tác cơ bản trên chuỗi string

- Kiểu **string** có thể chứa nội dung không giới hạn, vì đây là kiểu dữ liệu đối tượng được chứa ở bộ nhớ heap.
- Khai báo:

```
string s = "Nguyen Van A";
```

17

7. Thao tác cơ bản trên chuỗi ký tự

Ghép chuỗi: +

```
string a = "Xin";
```

```
string b = "chào";
```

```
string c = a + " " + b; // c = "Xin chào"
```

Hoặc

```
string.Concat(a, " ", b); → a = "Xin Chào"
```

18

7. Thao tác cơ bản trên chuỗi ký tự

Lấy chuỗi con: **Substring()**

string s;

s = "Lay chuoì con".Substring(4);

→ Lấy chuỗi con tính từ vị trí thứ 4 trở về sau:

→ s = "chuoì con"

s = "Lay chuoì con".Substring(4, 5);

→ Lấy chuỗi con từ vị trí thứ 4 và lấy chuỗi con có chiều dài là 5

→ s = "chuoì"

19

7. Thao tác cơ bản trên chuỗi ký tự

Thay thế chuỗi con

Replace (chuỗi cần thay, chuỗi thay thế)

string s;

s = "thay the chuoì.".Replace('t', 'T');

→ s = "Thay The chuoì"

s = "thay the chuoì.".Replace("th", "TH");

→ s = "THay THe chuoì"

20

7. Thao tác cơ bản trên chuỗi ký tự

Định dạng chuỗi: `Format` (định dạng, đổi số cần định dạng);

Ký tự	Mô tả	Ví dụ	Kết quả
C hoặc c	Tiền tệ (Currency)	<code>string.Format("{0:C}", 2.5);</code> <code>string.Format("{0:C}", -2.5);</code>	\$2.50 (\$2.50)
D hoặc d	Decimal	<code>string.Format("{0:D5}", 25);</code>	00025
E hoặc e	Khoa học (Scientific)	<code>string.Format("{0:E}", 250000);</code>	2.500000E+005
F hoặc f	Cố định phần thập phân (Fixed-point)	<code>string.Format("{0:F2}", 25);</code> <code>string.Format("{0:F0}", 25);</code>	25.00 25
G hoặc g	General	<code>string.Format("{0:G}", 2.5);</code>	2.5
N hoặc n	Số (Number)	<code>string.Format("{0:N}", 2500000);</code>	2,500,000.00
X hoặc x	Hệ số 16 (Hexadecimal)	<code>string.Format("{0:X}", 250);</code> <code>string.Format("{0:X}", 0xffff);</code>	FA FFFF

21

7. Thao tác cơ bản trên chuỗi ký tự

Chiều dài chuỗi: `Length`

```
string s = "Xin chào";
```

```
int n = s.Length; // n = 8
```

22

7. Thao tác cơ bản trên chuỗi ký tự

Tách chuỗi con theo ký hiệu phân cách cho trước

string.Split(danh sách ký tự phân cách)

Ví dụ:

In ra màn hình các từ trên từng dòng, mỗi từ cách nhau bằng dấu phẩy (,) hoặc khoảng trắng

```
string s = "Hom nay, ngay 02 thang 03 nam 2010";
foreach (string tu in s.Split(' ', ','))
    if (tu != "")
        Console.WriteLine(tu);
```

23

7. Thao tác cơ bản trên chuỗi ký tự

Ví dụ:

In ra từ có độ dài dài nhất trong chuỗi cho trước (từ cách nhau bằng khoảng trắng hoặc dấu chấm câu)

```
string s = "Hom nay, ngay 02 thang 03 nam 2010";
string tumax = "";
foreach (string tu in s.Split(' ', ',', '!', '?', ';'))
{
    if (tu != "" && tu.Length > tumax.Length)
        tumax = tu;
}
Console.WriteLine("Tu dai nhat: " + tumax);
```

24

8. Cấu trúc điều khiển

- Cấu trúc rẽ nhánh : if...else
- Cấu trúc lựa chọn : switch...case
- Cấu trúc lặp : for, while, do...while, foreach
- Các cấu trúc khác : goto, break, continue

25

8. Cấu trúc rẽ nhánh

```
if (Biểu thức điều kiện)
{
    // câu lệnh thực thi nếu Biểu thức điều kiện ĐÚNG
}
else
{
    // câu lệnh thực thi nếu Biểu thức điều kiện SAI
}
```

```
int x = 100;
if (x%2==0)
{
    Console.WriteLine("x chia hết cho 2");
}
else
{
    Console.WriteLine("x không chia hết cho 2");
}
```

26

8. Cấu trúc lựa chọn

switch (Biến điều kiện)

```
{
    case giá trị 1:
        Câu lệnh thực thi
        break;
    case giá trị 2:
        Câu lệnh thực thi
        break;
    case giá trị 3:
        Câu lệnh thực thi
        break;
    default:
        Câu lệnh thực thi
        break;
}
```

```
int x = 5;
switch (x)
{
    case 5:
        Console.WriteLine("Nam");
        break;
    case 10:
        Console.WriteLine("Muoi");
        break;
    default:
        Console.WriteLine("");
        break;
}
```

27

8. Cấu trúc lặp

while (biểu thức điều kiện)

```
{
    // câu lệnh
}
```

```
do
{
    // câu lệnh
}
while (biểu thức điều kiện)
```

for ([phần khởi tạo] ; [biểu thức điều kiện]; [bước lặp])

```
{
    // câu lệnh
}
```

foreach (<kiểu dữ liệu thành phần> <tên truy cập> in <mảng/tập hợp>)

```
{
    // thực hiện thông qua <tên truy cập> tương ứng với
    // từng mục trong mảng hay tập hợp
}
```

28

8. Cấu trúc lặp

```
int x = 10;

while (x > 0)
{
    Console.WriteLine("{0}", x);
    x--;
}

do
{
    Console.WriteLine("{0}", x);
    x--;
} while (x > 0);

for (int i = x; i > 0; i--)
    Console.WriteLine("{0}", i);

int[] arr = new int [10] {1,2,3,4,5,6,7,8,9,10};
foreach(int i in arr)
    Console.WriteLine("{0}", i);
```

29

8. Các lệnh break, goto và continue

```
int k = 0;
lap: // nhãn
    Console.WriteLine("k:{0}", k);
    k++;
if (k < 10)
    goto lap; // nhảy về nhãn lap
```

```
for (int i = 0; i < 10; i++)
{
    if (i % 2 == 0) continue;
    Console.WriteLine("i:{0}", i);
    if (i==7) break;
}
```

30

9. Phương thức và tham số

- **Mẫu tổng quát của phương thức:**

<Phạm vi> <KDL> TênPhươngThức ([Danh sách các tham số]);

- Phạm vi
 - Xác định phạm vi hay cách phương thức được gọi (sử dụng)
 - Các từ khoá phạm vi: private, public, static
- KDL của phương thức (đầu ra), gồm 2 loại
 - void: Không trả về giá trị
 - float / int / long / string / kiểu cấu trúc / ... : Trả về giá trị có KDL tương ứng với kết quả xử lý

31

9. Phương thức và tham số

- Tên phương thức : Đặt tên theo qui ước sao cho phản ánh đúng chức năng thực hiện của phương thức
- Danh sách các tham số (nếu có) : đầu vào của phương thức (trong một số trường hợp có thể là đầu vào và đầu ra của phương thức nếu kết quả đầu ra có nhiều giá trị - Tham số này gọi là tham chiếu)

32

9. Phương thức và tham số

- Phương thức không trả về giá trị

static void TênPhươngThức([danh sách các tham số])

{

 Khai báo các biến cục bộ

 Các câu lệnh hay lời gọi đến phương thức khác.

}

- Gọi hàm: TênPhươngThức(danh sách tên các đối số);
- Những phương thức loại này thường rơi vào những nhóm chức năng: Nhập / xuất dữ liệu, thống kê, sắp xếp, liệt kê

33

9. Phương thức và tham số

- Phương thức trả về giá trị

static <KDL> TênPhươngThức([tham số])

{

 <KDL> kq;

 Khai báo các biến cục bộ

 Các câu lệnh hay lời gọi đến phương thức khác.

 return kq;

}

- Gọi hàm: <KDL> Tên biến = TênPhươngThức(tên các đối số);
- Những phương thức này thường rơi vào các nhóm: Tính tổng, tích, trung bình, đếm, kiểm tra, tìm kiếm

34

BÀI TẬP

Bài 1: Thực hiện các yêu cầu sau trên giao diện winform

- Tính tiền grab theo bảng giá sau:

Km	Giá / km
5km đầu tiên	20.000
15km tiếp theo	15.000
Các km tiếp theo	10.000
Nếu đi trên 100km thì được giảm giá 10%	

→ Nhập số km đi được và tính tiền grab

35

BÀI TẬP

Bài 2: Yêu cầu thiết kế giao cho "Trang chủ" và "Giỏ hàng" với các chức năng sau:

- Khi click vào "Trang chủ" (biểu tượng Home) sẽ hiển thị giao diện như hình bên dưới:
- Khi click vào từng dịch vụ để chọn và hủy chọn dịch vụ. Dịch vụ được chọn sẽ thay đổi màu nền.



36

BÀI TẬP

- Khi click vào "Giỏ hàng" (biểu tượng giỏ hàng) sẽ hiển thị giao diện như hình bên dưới:
- Các dịch vụ được chọn sẽ hiển thị tương ứng số tiền theo bảng giá dịch vụ ở mô tả phần 2

Bảng giá dịch vụ:

Giặt ủi: 100.000
 Vệ sinh máy lạnh: 150.000
 Dọn dẹp: 200.000
 Tổng vệ sinh: 400.000
 Nấu ăn: 300.000
 Dịch vụ khác: 500.000

37

BÀI TẬP

Bài 3: Thực hiện các yêu cầu sau trên giao diện winform

- Quản lý khách sạn gồm các thông tin:

Thông tin	Kiểu dữ liệu	Ghi chú
Mã phòng	string	VD: A1, A2, A3, B1, B2, C1...
Loại phòng	string	Có 3 loại: A, B, C
Giá phòng	int	Giá phòng trên 1 ngày cho mỗi loại
Tình trạng	int	không trống – trống

- Thêm phòng
- Đặt phòng → Chọn phòng, số ngày ở
- Hủy đặt phòng
- Tính tiền phòng dựa theo số ngày ở

Loại phòng	Giá phòng
A	150.000
B	200.000
C	300.000

38