枚举、结构体和数组

**枚举**

enum WeekDays {Sunday,Monday,Tuesday,Wednesday,Thursday,Friday,Saturday}

与C++一样，枚举类型缺省初始值也是0，也可以强制转换为整形。也可以使用conslole.writeline输出。

**结构体**

//定义结构体

struct Student

{

public int number;

public string name;

public string sex;

public string birthday;

}

**5.3 数组**

数组初始化 int[] score = { 68, 97, 45, 67 };

string[] friendnames = new string[3];

**5.3.2 foreach循环语句**

string[] friendNames = { "张三", "李四", "王五" };

Console.WriteLine("我共有{0}位朋友：", friendNames.Length);

foreach (string name in friendNames)

{

Console.WriteLine(name);

}

**5.3.3 二维数组**

int[,] Matrix = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 } };

二维数组中的每个一维数组的元素个数应该一致，否则无法编译通过。

int[,] Matrix2 = new int[3, 3];

**5.3.4 可变数组**

int[][] jaggedArray = new int[2][];

声明可变数组需要指定行数，但不需要指定列数。初始化这样的数组不像初始化二维

数组那样简单，需要逐行初始化。

jaggedArray[0] = new int[3]; //初始化第一行，3 个元素

jaggedArray[1] = new int[4]; //初始化第二行，4 个元素

输出可变数组

foreach (int[] line in jaggedArray)

{

foreach (int element in line)

{

Console.Write(element);

}

}