**C#语法**

1. **修饰符**

Const //创建常量

partial //在不同文件创建同一类结构

——————————————————————————

public 公开

private 私有

protected 保护

1. **类型**

enum{，，，} //枚举

interface //接口

class //类（引用类型）

sealed //密封的

abstract //抽象的

struct //结构（值类型）

非引用类型+？ //使其可空null

1. **泛型**

void 函数名<类型>(类型 变量)

//无法确定类型时使用，调用时会自动检测

//泛型名自定义，比用Object优化好

class<泛型>

where 类型:限制 //限定泛型

{} //泛型类，成员可使用

//创建对象时应指明类型

1. **强制转换类型**

Convert.To+(类型)( 变量)

(类型).Parse(变量) //指定数据隐式转换指定类型

(类型).TryParse(变量,out 变量) //隐式转换指定数据至变量

1. **方法**

delegate //委托(方法组)

virtual //虚拟的

override //推翻

ref/out //传递实参

params //将参数自动转发为形参数组

——————————————————————————

public static 返回类型 operator（必须有一个参数为所在类的类型）{}

//重载运算符

1. **字符处理**

(string变量).ToUpper()

(string变量).TOLower()

//大小写转换

(string变量).Trim() //删除两边空格

(string变量).Replace() //替换指定字符

(string变量).Substring() //截取字段

(string变量).Split() //拆分字段

string.Compare() //比较字符是否一样

1. **数组处理**

ArrayList //列表

Hashtable //泛型列表

Dictionary //字典

Array.Sort()//从小到大排序

Array.Reverse()//从大到小排序

Array.Resize(ref)//调整数组大小

——————————————————————————

foreach(类型 变量名 in 数组/集合){ }

//搜索数组

1. **文件处理**

IO文件处理库

+Info //实例化

Directory //目录

File //文件

FileStream //文件数据

MemoryStream //内存文件流

Encoding //编码解码

1. **多线程**

using System.Threading //声明多线程类库

//多线程方法只能带Object参数

Thread.CurrentThread //获取当前线程

Thread.IsBackground //是否为后台线程

Thread.Priority=int //设置线程优先级

Thread.start() //启动线程

Thread.sleep() //当前线程暂停

volatile 值类型 //易失域：在多线程中保持绝对

lock（this）{} //锁定代码，同时只能有一个线程处理

1. **网络**

Using System.Net //网络类库

Using System.Net.Sockets //Socket类库

Socket（寻址方案，传输类型，传输协议） //创建网络接口

IPEndPoint(IPAddress+., 端口); //创建网络地址

服务端

Bind (地址) //创建服务器

Listen(int) //设置队列长度

=Accept() //开始监听连接

客户端

Connect(地址) //连接服务器

通用操作

Send(byte()) //发送数据

Receive(byte()) //接受数据

1. **特性反射**

[AttributeUsage(AttributeTargets.All,AllowMultiple = true,Inherited = true)]

//[特性属性(使用范围，能否多次使用，能否继承)]

public class 名称+Attribute : Attribute{}

//自定义特性

——————————————————————————

[Obsolete(string,bool)] 过时特性

——————————————————————————

System.Reflection 反射类库

1. **高级**

# //预处理

#region 名称+#endregion

//编辑器中创建折叠区域

——————————————————————————

unsafe class 类名{} //不安全类型或成员

unsafe{} //不安全代码块

\*值 &地址

fixed (int\* a = &b){} //指针赋值程序块

sizeof //计算非托管类型的大小

——————————————————————————

CG.Collect()

//空对象回收

1. **异常**

throw new 异常类型 //产生一个异常

try{} //异常语句块

catch{ throw； } //将处理的异常再次抛出

——————————————————————————

Checked //检测溢出异常

checked{unchecked{}} //即使溢出相对checked也不报错

1. **其他**

Equals //比较Object是否相同

——————————————————————————

XML文件处理库

——————————————————————————

using Microsoft.VisualBasic.Devices;

using Microsoft.VisualBasic;

//VB的便捷函数库C#可以借用