MFC和COM编程技术：老式编程环境  
MFC（Microsoft foundation classes，微软基础类库）  
COM（Component Object Model，组件对象模型）

managed code：托管代码，也称IL中间语言，是所有面向CLR兼容的编译器编译时产生的中间模块的一部分。

managed module：托管模块，包括IL和元数据。是CLR兼容编译器产生的中间文件。

PE（portable executable）：可移植可执行文件，托管模块的文件类型，需要CLR才能执行。

平台调用(platform invoke，P/Incoke)

CLI（Common Language Infrastructure）：公共语言基础架构，是.NET技术的完整规范，包括——  
CTS（Common Type System）：公共类型系统，定义一组内置类型  
CLS（Common Language Specification）：公共语言规范，说明兼容.NET的编程语言的规则  
CIL（Common Intermediate Language）：公共中间语言  
CLR（Common Language Runtime）：公共语言运行库，程序运行时的库/引擎  
BCL（Base Class Library）：基类库，FCL的子集  
元数据定义和语义  
其他相关标准化文档、协议和规范

.NET Framework是CLI在Windows上的实现，包括——  
CLR：提供一系列运行时服务和其他功能的核心组件  
FCL（Framework Class Library）：框架类库

GC（Garbage Collection）：自动垃圾收集，清理内存

CLR在运行期管理程序的执行，包括以下内容：  
1）内存管理。  
2）代码安全验证。  
3）代码执行、线程管理和异常处理。  
4）垃圾收集。

CLR提供的服务：  
1）内存管理。  
2）异常处理。  
3）自动垃圾收集。  
4）管理代码的执行  
5）JIT实时编译器（Just-in-Time）。  
6）类加载器。【通过访问BCL得到广泛的编程功能，如web服务和数据服务之类的功能】  
7）反射服务。  
8）安全服务。  
9）提供系统服务。

FCL的组成 或 服务：  
1）基础类型。【整数、实数、字符串等类型定义】  
2）数据结构封装。【集合、链表、队列、堆栈等数据类型】  
3）Windows和Web等界面要素。【主要是按钮、标签、文本框、菜单等可视化控件】  
4）Web Service要素。【用于Web服务的定义、描述、配置、解析等】  
5）XML文档处理。【如XML文件、属性、元素、节点、读写器、解析器等类型】  
6）文件的输入输出。【如驱动器、目录、文件、流、读写器等类型】  
7）数据访问。【如数据连接、数据命令、数据集、数据表、记录等类型】  
8）网络通信。【如主机、管道、套接字、消息等类型】  
9）异常处理。【用于处理系统和应用程序所引发的各种异常】  
9）类型反射。【用于获取程序集、对象、方法属性、字段等目标的元数据信息类型】  
10）用于应用程序管理、操作系统功能封装、安全性控制等其他方面的类型。