一．选择，填空题

1. 在ADO.NET中，对于Command对象的ExecuteNonQuery()方法和ExecuteReader()方法，下面叙述错误的是（C）。
   1. insert、update、delete等操作的Sql语句主要用ExecuteNonQuery()方法来执行；
   2. ExecuteNonQuery()方法返回执行Sql语句所影响的行数。
   3. Select操作的Sql语句只能由ExecuteReader()方法来执行；~
   4. ExecuteReader()方法返回一个DataReder对象；
2. 面向对象的语言具有\_\_继承性\_性、\_封装性\_性、\_多态性 性。
3. 能用foreach遍历访问的对象需要实现 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_接口或声明\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_方法的类型。(@)

答：IEnumerable 、 GetEnumerator()

1. 在C#中,表示一个字符串的变量应使用以下哪条语句定义？( B )

A. CString str;

B. string str;

C. Dim str as string

D. char \* str;

1. 在C#编制的财务程序中,需要创建一个存储流动资金金额的临时变量,则应使用下列哪条语句？( A )

A. decimal theMoney;

B. int theMoney;

C. string theMoney;

D. Dim theMoney as double

1. C#中,新建一字符串变量str,并将字符串”Tom's Living Room”保存到串中,则应该

使用下列哪条语句？(B)

A. string str = “Tom\'s Living Room”;

B. string str = “Tom's Living Room”;

C. string str(“Tom's Living Room”);

D. string str(“Tom”s Living Room”);

1. 应用ADO.NET 访问数据时,Connection 对象的连接字符串中Initial Catalog 子串的

含义是 ( A )

A. Connection 对象连接到的数据库的名称

B. Connection 对象的身份验证信息

C. Connection 对象的最大连接时间

D. Connection 对象使用的缓存大小

1. 使用Visual Studio .NET 的”新建C# 项目”创建一个名为”SimpleForm”的Windows

表单应用程序,则在生成\_\_\_\_\_\_\_文件中可设置该程序集的Copyright、Trademark 等属

性信息.（ D ）

A. Form1.resx

B. SimpleForm.sln

C. SimpleForm.csproj

D. AssemblyInfo.cs

1. C#程序中,为使变量myForm 引用的窗体对象显示为对话框,必须: （ C ）

A. 使用myForm.ShowDailog 方法显示对话框

B. 将myForm 对象的isDialog 属性设为true

C. 将myForm 对象的FormBorderStyle 枚举属性设置为FixedDialog

D. 将变量myForm 改为引用System.Windows.Dialog 类的对象

1. 下列哪些语言可以用来开发XML Web Service? （ A，B，C，D ）

A. C

B. C++

C. Java

D. C#

1. Web 表单中,按钮的默认事件是 （ A ）

A. Click 事件

B. Load 事件

C. Init 事件

D. Command 事件

1. 关于ASP.NET 窗体应用程序,下列说法哪些是正确的？ （ D ）

A. Web 窗体是在浏览器中进行解释执行的

B. Web 窗体程序中脚本和代码必须严格分开

C. Web 窗体程序产生的HTML 页面只能运行于IE 浏览器

D. Web 窗体程序的代码可以使用任何一种.NET 兼容的编程语言编写

1. 关于结构类型,下列说法哪些是正确的？ （ A ）

A. 结构是值类型

B. 结构中不允许定义带参数的实例构造函数;

C. 结构中不允许定义析构函数

D. 结构中可以定义成员方法,但是方法内不能使用this 指针

E. 不能使用new 关键字创建结构类型的实例

1. 关于线程和进程,下列说法哪些是正确的？ （ A D ）

A. 线程是操作系统分配处理器时间的基本单位;

B. 进程是操作系统分配处理器时间的基本单位;

C. 一个线程可以属于多个进程;

D. 一个进程可以有多个线程;

1. 为从某个网站访问xml信息,应使用( \_)类的对象分析下载返回的xml文档 (C D)

A StreamReader

B TextReader

C XmlReader

D XmlTextReader

1. 面向对象编程中的”继承”的概念是指(B)

A 派生类对象可以不受限制地访问所有的基类对象

B 派生自同一个基类的不同类的对象具有一些共同特征

C 对象之间通过消息进行交互

D 对象的内部细节被隐藏

1. 以下叙述正确的是（ D ）。@

A. 一条表达式中可以同时出现浮点数类型和decimal类型

B. 类可以继承，并且可以多重继承

C. int[,] b=new int[4,5] //创建了一个二维数组，第一维长为4，第二维长为5

D. int x=23; object o=x; int y=(int) o; //实现了一次装箱操作

1. 以下有关结构类型和枚举类型的叙述错误的是（ C　）。

A. 结构类型和枚举类型都是复合类型

B. 结构类型只能是简单值类型的组合

C. 枚举类型要求成员只能是整数类型

D. 枚举类型的变量在某一时刻只

1. 接口和类最主要的区别在于（ D ）。

A．类可以重载而接口不可以

B．类不可以重载而接口可以

C．类可以多重重载而接口不可以

D．类不可以多重重载而接口可以

1. 关于静态域成员与非静态域成员，以下说法正确的是（ C ）。

A. 一个类的所有对象共享非静态域成员

B. 一个类的一个对象分别拥有不同的非静态域成员的实例

C. 静态域成员都能在不生成实例的情况下直接被使用

D. 非静态域成员都能

1. 关于重载，以下叙述错误的是（C，D）。

A. 重载是指在类的内部存在若干个方法名称一致但是参数列表不同的方法

B. 方法重载是指创建多个名称相同的方法，其中每个方法都在某些方面具有唯一性，以便编译器能够正确区别它们

C. 重载构造函数主要是为了在创建对象的同时进行赋值

D. 方法的重载允许两个方法的名称和参数列表相同，而返回类型不同

1. 下面有关C＃中方法的参数描述正确的是（B ）。

A. 利用值类型参数调用方法时，对形参的修改会反映到实参上去

B. 输出型参数的实参在传递给形参前，不需要明确赋值

C. 引用型参数不另外开辟新的内存区域，这一点是它与输出型参数的不同之处

D. 数组型参数可以是任何类型的数组

1. 下面是一个派生类的定义语句：

public class car: vehicle

{ private string model=”L”;

private int wheelNo=4;}

请指出针对该定义的正确的说法（C ）。

A. vehicle类是car类的一个特例，因而它可以被继承

B. car类是父类，vehicle类是子类

C. 在car类中定义的成员model和wheelNo会删除vehicle类中同名的成员

D. car类是vehicle类的一个特例，它可以继承vehicle类中除构造函数和析构函

1. （D ）关键字可以用来访问当前对象的基类对象。

A．object B．this

C．as D．base

1. 重载是（B ）的形式之一。

A．接口 B．多态

C．继承 D．类

1. x的初值为24，则表达式x+=x--+=x%=17运算的结果值为（D ）。

A. 28 B．27

C．26 D．错误的表达式

1. 下面是几条定义初始化二维数组的语句，指出其中正确的是（D ）。

A．int array2[ ][ ]=new int[3,2];

B．int array2[,]=new int[3,2];

C．int[ ][ ] array2=new int[3,2];

D．int[, ] array2=new int[3,2];

1. 一个初始化语句：

enum monthes{January,February=3,March=5,April=7,May,June},则此时May的值为（C ）。

A．4 B．5 C．8 D．9

1. 在Asp.net中所有的自定义用户控件都必须继承自\_\_\_\_\_\_\_\_? 答：Control。
2. 在.Net中所有可序列化的类都被标记为\_\_\_\_\_?

答：[serializable]

1. 在.Net托管代码中我们不用担心内存漏洞，这是因为有了\_\_\_\_\_\_?

答：GC。

1. 在.Net中，类System.Web.UI.Page 可以被继承么？

答：可以。

1. 声明静态变量的关键字是（static），声明常量的关键字是（const）
2. 在流程控制中CONTINUE的作用（跳出当前循环继续下一次循环）
3. 软件工程的三大文档（需要分析说明书）（概要设计说明书）（详细设计说明书）。
4. B/S通常使用的结构设计模型分为（模型）（视图）（控制器）。（MVC经典的Web/Business/DataAccess）
5. XML的全称是（Extensible Markup Language”可扩展标记语言”），提供了快速和有效地读写XML的方式的类分别是（XMLReader）（XMLWriter）
6. C#中所有的类型实质上都是从（Object）类派生而来的。
7. NET运行库支持被称为（委托）的引用类型，其作用类似于C++中函数指针的用途。
8. B/S通常使用的结构设计模型分为（模型）（视图）（控制）?\*
9. 类的三大特性是（封装）（继承）（多态）
10. .NET运行库支持被称为（委托）的引用类型，其作用类似于C++中函数指针的用途
11. C#中处理内存管理功能的是（托管代码GC(垃圾回收)）
12. 在C#中用于显式地将一个值或引用转换为另一种不同的类型称为（拆箱）
13. 在ADO。NET中，command对象是用（ExcuteNonQuery）（ExcuteScalar）（ExcuteDataReader）方法执行命令

**二．简答题**

1. 简述 private、 protected、 public、 internal 修饰符的访问权限。

答：private : 私有成员, 在类的内部才可以访问。

protected : 保护成员，该类内部和继承类中可以访问。

public : 公共成员，完全公开，没有访问限制。

internal: 在同一命名空间内可以访问。

2. C#中，string str = null 与 string str =""，请尽量用文字说明区别。

答：string str ="" 分配空间而string str = null不分配空间。

3.用sealed修饰的类有什么特点？

答：不可被继承

?\*4.请叙述const与readonly的区别。

* 1. 每一个class至多只可以定义一个static构造函数，并且不允许增加访问级别关键字，参数列必须为空。  
              为了不违背编码规则，通常把static数据成员声明为private，然后通过statci property提供读写访问。  
              const 关键字用于修改字段或局部变量的声明。它指定字段或局部变量的值不能被修改。常数声明引入给定类型的一个或多个常数。  
              const数据成员的声明式必须包含初值，且初值必须是一个常量表达式。因为它是在编译时就需要完全评估。  
              const成员可以使用另一个const成员来初始化，前提是两者之间没有循环依赖。  
              readonly在运行期评估赋值，使我们得以在确保“只读访问”的前提下，把object的初始化动作推迟到运行期进行。  
              readonly 关键字与 const 关键字不同：　const 字段只能在该字段的声明中初始化。readonly 字段可以在声明或构造函数中初始化。因此，根据所使用的构造函数，readonly 字段可能具有不同的值。另外，const 字段是编译时常数，而 readonly 字段可用于运行时常数。   
     　　readonly 只能在声明时或者构造函数里面初始化，并且不能在 static 修饰的构造函数里面。

5.简述 private、 protected、 public、 internal 修饰符的访问权限。

private : 私有成员, 在类的内部才可以访问。

protected : 保护成员，该类内部和继承类中可以访问。

public : 公共成员，完全公开，没有访问限制。

internal: 在同一命名空间内可以访问。

**?\*6.请说明在.net中常用的几种页面间传递参数的方法，并说出他们的优缺点。**session(viewstate) 简单，但易丢失 @  
application 全局   
cookie 简单，但可能不支持，可能被伪造   
input ttype="hidden" 简单，可能被伪造   
url参数简单，显示于地址栏，长度有限   
数据库稳定，安全，但性能相对弱

**7.概述反射和序列化**  
答:反射:程序集包含模块，而模块包含类型，类型又包含成员。反射则提供了封装程序集、模块和类型的对象。您可以使用反射动态地创建类型的实例，将类型绑定到现有对象，或从现有对象中获取类型。然后，可以调用类型的方法或访问其字段和属性  
序列化:序列化是将对象转换为容易传输的格式的过程。例如，可以序列化一个对象，然后使用 HTTP 通过 Internet 在客户端和服务器之间传输该对象。在另一端，反序列化将从该流重新构造对象。

8.什么是SOAP,有哪些应用。

答:SOAP（Simple Object Access Protocol ）简单对象访问协议（SOAP）是一种轻量的、简单的、基于 XML 的协议，它被设计成在 WEB 上交换结构化的和固化的信息。 SOAP 可以和现存的许多因特网协议和格式结合使用，包括超文本传输协议（ HTTP），简单邮件传输协议（SMTP），多用途网际邮件扩充协议（MIME）。它还支持从消息系统到远程过程调用（RPC）等大量的应用程序。Web service 就是一个应用程序，它向外界暴露出一个能够通过Web进行调用的API。这就是说，你能够用编程的方法通过Web来调用这个应用程序.

9.ASP。NET中共有几种类型的控件？各有什么区别？

答:Html 控件传统的html标记     
 Web 控件可以回传数据，事件驱动     
 自定义控件在原有控件的基础上增加功能     
 复合控件多个子控件复合成一个新的控件

 10.什么情况用HTML控件，什么情况用WEB控件，并比较两者差别

答:客户端表现用html控件，如果想和server端交互，那么加上runat=server，它就成了服务器端控件，但它没有web控件的很多方法和属性，如果你需要用到，那么还是用webControls.

11.用.net做B/S结构的系统，您是用几层结构来开发，每一层之间的关系以及为什么要这样分层？  
答：一般为3层  
数据访问层，业务层，表示层。  
数据访问层对数据库进行增删查改。  
业务层一般分为二层，业务表观层实现与表示层的沟通，业务规则层实现用户密码的安全等。  
表示层为了与用户交互例如用户添加表单。  
优点： 分工明确，条理清晰，易于调试，而且具有可扩展性。  
缺点： 增加成本。

12.什么是XML？  
答：XML即可扩展标记语言。eXtensible Markup Language.标记是指计算机所能理解的信息符号，通过此种标记，计算机之间可以处理包含各种信息的文章等。如何定义这些标记，即可以选择国际通用的标记语言，比如HTML，也可以使用象XML这样由相关人士自由决定的标记语言，这就是语言的可扩展性。XML是从SGML中简化修改出来的。它主要用到的有XML、XSL和XPath等。

**13.什么是Web Service？UDDI？@**  
答：Web Service便是基于网络的、分布式的模块化组件，它执行特定的任务，遵守具体的技术规范，这些规范使得Web Service能与其他兼容的组件进行互操作。  
UDDI 的目的是为电子商务建立标准；UDDI是一套基于Web的、分布式的、为Web Service提供的、信息注册中心的实现标准规范，同时也包含一组使企业能将自身提供的Web Service注册，以使别的企业能够发现的访问协议的实现标准

**14.WEB控件可以激发服务端事件，请谈谈服务端事件是怎么发生并解释其原理？自动传回是什么？为什么要使用自动传回。**

在web控件发生事件时，客户端采用提交的形式将数据交回服务端，服务端先调用Page\_Load事件,然后根据传回的状态信息自动调用服务端事件自动传回是当我们在点击客户端控件时，采用提交表单的形式将数据直接传回到务端

只有通过自动传回才能实现服务端事件的机制，如果没有自动回传机制就只能调用客户端事件，而不能调用服务端事件

**16.请简述DataReader与DataSet的异与同。**

**17.PDB是什么东西? 在调试中它应该放在哪里？**

答：PDB是用于保存调试和项目状态信息的文件，在debug的时候将产生pdb文件，调试的时候应该放在和对应应用程序集相同目录。

**18.简述ajax的原理。**

答:ajax简单的讲就是客户端通过javascript脚本获取服务器端的文本，通过解析返回值，更新部分的网页内容。

**19.描述线程与进程的区别？**

答:线程(Thread)与进程（Process）二者都定义了某种边界，不同的是进程定义的是应用程序与应用程序之间的边界，不同的进程之间不能共享代码和数据空间，而线程定义的是代码执行堆栈和执行上下文的边界。一个进程可以包括若干个线程，同时创建多个线程来完成某项任务，便是多线程。而同一进程中的不同线程共享代码和数据空间。用一个比喻来说，如果一个家庭代表一个进程，在家庭内部，各个成员就是线程，家庭中的每个成员都有义务对家庭的财富进行积累，同时也有权利对家庭财富进行消费，当面对一个任务的时候，家庭也可以派出几个成员来协同完成，而家庭之外的人则没有办法直接消费不属于自己家庭的财产。

**20.什么是Windows服务，它的生命周期与标准的EXE程序有什么不同**

答:Windows服务是运行在windows后台指定用户下（默认System）的应用程序，它没有标准的UI界面，想比标准的EXE程序，Windows服务是在服务开始的时候创建，而在服务结束的时候销毁，而且可以设置服务是否与操作系统一起启动，一起关闭。它支持三种方式：１）自动方式 2）手动方式　３）禁用 。自动方式的时候，windows服务将在OS启动后自动启动运行，而手动方式则必须手工启动服务，禁用的情况下服务将不能被启动。另外标准的EXE默认使用的当前登录的用户，而windows服务则默认使用System用户，这在对系统资源访问的时候特别需要注意。

**21.什么是强类型，什么是弱类型？哪种更好些？为什么?**

答:强类型是在编译的时候就确定类型的数据，在执行时类型不能更改，而弱类型在执行的时候才会确定类型。没有好不好，二者各有好处，强类型安全，因为它事先已经确定好了，而且效率高。一般用于编译型编程语言，如c++,java,c#,pascal等,弱类型相比而言不安全，在运行的时候容易出现错误，但它灵活，多用于解释型编程语言，如javascript,vb等 .

**22.什么是GAC？它解决了什么问题?**

答:Gloal Assembly Cache，全局应用程序集缓存。它解决了几个程序共享某一个程序集的问题。不必再将那个被共享的程序集拷贝到应用程序目录了，其实这道理很简单，.net应用程序在加载的时候，会首先查看全局应用程序集缓存，如果有就可以直接使用，没有再到应用程序目录进行查找。

**23.阐述面向接口、面向对象、面向方面编程的区别**

答:面向接口更关注的是概念，它的原则是先定义好行为规范，再根据行为规范创建实现，严格的来说，面向接口应该是面向对象中的一部分吧，因为面向对象也强调的是本末倒置原则，也就是实现依赖于抽象，而抽象不依赖于具体实现，更具比较的应该是面向接口与面向抽象对象，我的体会是面向接口更加灵活，但实现时候，稍微有些代码冗余，而面向抽象可以结合面向接口，先定义接口，再定义抽象类，在抽象类中处理一些公共逻辑，再实现具体实现类。面向对象是对复杂问题的分解。面向方面的编程是一种新概念，它解决了很多面向对象无法解决的问题，比如面向对象技术只能对业务相关的代码模块化，而无法对和业务无关的代码模块化。而面向方面正是解决这一问题的方案，它的关键思想是"将应用程序中的商业逻辑与对其提供支持的通用服务进行分离"。

**24.简述用repeater实现手工分页的思路。**

**25.web service 数据传输有什么限制? 为什么?DataTable可以作为web service参数传递么?**

答:所传输的数据必须是可序列化的。因为需要转换为XML格式以可以穿越防火墙，做到真正的数据共享。因为DataSet处理DataTable的序列化以便进行封送处理，所以无法传递单个DataTable

**26.常用的调用WebService的方法有哪些？**

答:在同一个解决方案中调用WebService方法；   
在不同的解决方案中调用WebService方法；   
在Internet上调用WebService方法；   
利用webservice.htc调用WebService方法。

**28.在C＃中，string str = null 与 string str = “” 请尽量使用文字或图象说明其中的区别。**

答：string str = null 是不给他分配内存空间,而string str = "" 给它分配长度为空字符串的内存空间.

**30.请详述在dotnet中类(class)与结构(struct)的异同：**   
答：Class可以被实例化,属于引用类型,是分配在内存的堆上的,Struct属于值类型,是分配在内存的栈上的.

**31. 如何理解委托？**

答:据说相当于函数指针,定义了委托就可以在不调用原方法名称的情况下调用那个方法.

msdn2005中是这样解释的:

委托具有以下特点：

委托类似于 C++ 函数指针，但它是类型安全的。

委托允许将方法作为参数进行传递。

委托可用于定义回调方法。

委托可以链接在一起；例如，可以对一个事件调用多个方法。

方法不需要与委托签名精确匹配。有关更多信息，请参见协变和逆变。

C# 2.0 版引入了匿名方法的概念，此类方法允许将代码块作为参数传递，以代替单独定义的方法。

**33． override与重载的区别**

override 与重载的区别。重载是方法的名称相同。参数或参数类型不同，进行多次重载以适应不同的需要

Override 是进行基类中函数的重写。为了适应需要。

**34. abstract class和interface有什么区别?**

　　答：声明方法的存在而不去实现它的类被叫做抽象类(abstract class)，它用于要创建一个体现某些基本行为的类，并为该类声明方法，但不能在该类中实现该类的情况。不能创建abstract 类的实例。然而可以创建一个变量，其类型是一个抽象类，并让它指向具体子类的一个实例。不能有抽象构造函数或抽象静态方法。Abstract 类的子类为它们父类中的所有抽象方法提供实现，否则它们也是抽象类为。取而代之，在子类中实现该方法。知道其行为的其它类可以在类中实现这些方法。

接口(interface)是抽象类的变体。在接口中，所有方法都是抽象的。多继承性可通过实现这样的接口而获得。接口中的所有方法都是抽象的，没有一个有程序体。接口只可以定义static final成员变量。接口的实现与子类相似，除了该实现类不能从接口定义中继承行为。当类实现特殊接口时，它定义(即将程序体给予)所有这种接口的方法。然后，它可以在实现了该接口的类的任何对象上调用接口的方法。由于有抽象类，它允许使用接口名作为引用变量的类型。通常的动态联编将生效。引用可以转换到接口类型或从接口类型转换，instanceof 运算符可以用来决定某对象的类是否实现了接口。

**35. Session有什么重大BUG，微软提出了什么方法加以解决?**

答：是iis中由于有进程回收机制，系统繁忙的话Session会丢失，可以用Sate server或SQL Server数据库的方式存储Session不过这种方式比较慢，而且无法捕获Session的END事件。

**36. 成员变量和成员函数前加static的作用?**

答：它们被称为常成员变量和常成员函数，又称为类成员变量和类成员函数。分别用来反映类的状态。比如类成员变量可以用来统计类实例的数量，类成员函数负责这种统计的动作。

**37.简述三层架构的利弊。**

**38.简述AJAX控件timer的实现原理。**

**39.简述ASP.NET中用户控件和自定义控件的区别.**

**40.简述framework在.NET中的作用.**

**41.简述C#中引入泛型的好处。**

**42.谈谈arraylist和list<>利弊.**

**43.怎样实现用JS调用服务器端定义的方法。**

**44.简述ADO.NET中几大对象的作用.**

**45.相对与其它WEB开发平台，微软的.net平台有什么独特的优势和劣势。**

**46.谈谈Schema和DTD相比，具有那些显著的优势。**

**47. 什么叫应用程序域？什么是受管制的代码？什么是强类型系统？什么是装箱和拆箱？什么是重载？CTS、CLS和CLR分别作何解释？**

答: （1）应用程序域（通常为 AppDomain）是一个用于隔离应用程序的虚拟进程。

（2） 受管制的代码：Managed Code，官方翻译是“托管代码”，即在 .net中，由公共语言运行环境进行管理的代码，它主要负责内存管理。

（3）强类型系统：是对系统的类型进行细化定义，纯粹是概念问题，不做深入研究的话，看书理解一下就行。

（4）装箱和拆箱：值类型转化引用类型为装箱，引用类型转化为值类型为拆箱。

（5）CTS：通用语言系统。CLS：通用语言规范。CLR：公共语言运行库。

**48.强类型与弱类型的区别？**

（1）答:完全的面向对象要求对象是强类型的。强类型是只在对象类型的定义尽量的细化。我们可以定义所有的对象都是object,然后允许后期绑定来对对象进行操作。事实上，有些语言是弱类型的，特别是一些教本语言，比如javascript.弱类型语言对简化了语言本身，因而语言本事是可以轻量的，在一些特殊的场合还是很有用的。

（2）但弱类型与强类型语言相比，它们能完成的功能从本质上来说，是没有区别的。但弱类型的轻量化是以牺牲开发的方便性和效率为基础的。它将一部分机器能完成的事情移到程序员来完成。如果开发同样一个功能，如果用javascript 和强类型的vb.net来开发，两者对程序员的要求与开发所需要的时间是明显不同的。

**49.请详述在c#中类(class)与结构(struct)的异同？**

答：Class可以被实例化,属于引用类型,是分配在内存的堆上的,Struct属于值类型,是分配在内存的栈上的. Class可以被继承，struct不可以等等

**50. GC是什么? 为什么要有GC?**

答：GC是垃圾收集器。程序员不用担心内存管理，因为垃圾收集器会自动进行管理。要请求垃圾收集，可以调用下面的方法之一：

System.gc()

Runtime.getRuntime().gc()

**51.String s = new String("xyz");创建了几个String Object?**

答：两个对象，一个是“xyx”,一个是指向“xyx”的引用对象s。

**52.abstract class和interface有什么区别?**

答：声明方法的存在而不去实现它的类被叫做抽象类（abstract class），它用于要创建一个体现某些基本行为的类，并为该类声明方法，但不能在该类中实现该类的情况。不能创建abstract 类的实例。然而可以创建一个变量，其类型是一个抽象类，并让它指向具体子类的一个实例。不能有抽象构造函数或抽象静态方法。Abstract 类的子类为它们父类中的所有抽象方法提供实现，否则它们也是抽象类为。取而代之，在子类中实现该方法。知道其行为的其它类可以在类中实现这些方法。

接口（interface）是抽象类的变体。在接口中，所有方法都是抽象的。多继承性可通过实现这样的接口而获得。接口中的所有方法都是抽象的，没有一个有程序体。接口只可以定义static final成员变量。接口的实现与子类相似，除了该实现类不能从接口定义中继承行为。当类实现特殊接口时，它定义（即将程序体给予）所有这种接口的方法。然后，它可以在实现了该接口的类的任何对象上调用接口的方法。由于有抽象类，它允许使用接口名作为引用变量的类型。通常的动态联编将生效。引用可以转换到接口类型或从接口类型转换，instanceof 运算符可以用来决定某对象的类是否实现了接口。

**53.启动一个线程是用run()还是start()?**

答：启动一个线程是调用start()方法，使线程所代表的虚拟处理机处于可运行状态，这意味着它可以由JVM调度并执行。这并不意味着线程就会立即运行。run()方法可以产生必须退出的标志来停止一个线程。

**54.接口是否可继承接口? 抽象类是否可实现(implements)接口? 抽象类是否可继承实体类(concrete class)?**

答：接口可以继承接口。抽象类可以实现(implements)接口，抽象类是否可继承实体类，但前提是实体类必须有明确的构造函数。

**55.构造器Constructor是否可被override?**

答：构造器Constructor不能被继承，因此不能重写Overriding，但可以被重载Overloading。

**56.是否可以继承String类?**

答：String类是final类故不可以继承。

**57. try {}里有一个return语句，那么紧跟在这个try后的finally {}里的code会不会被执行，什么时候被执行，在return前还是后?**

答：会执行，在return前执行。

**58.两个对象值相同(x.equals(y) == true)，但却可有不同的hash code，这句话对不对?**

答：不对，有相同的hash code。

**59.swtich是否能作用在byte上，是否能作用在long上，是否能作用在String上?**

答：可以

**60.当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法?**

答：不能，一个对象的一个synchronized方法只能由一个线程访问。

**61.abstract的method是否可同时是static,是否可同时是native，是否可同时是synchronized?**

答：都不能。

**62.List, Set, Map是否继承自Collection接口?**

答：List，Set是Map不是

**63. Set里的元素是不能重复的，那么用什么方法来区分重复与否呢? 是用==还是equals()? 它们有何区别?**

答：Set里的元素是不能重复的，那么用iterator()方法来区分重复与否。equals()是判读两个Set是否相等。

equals()和==方法决定引用值是否指向同一对象equals()在类中被覆盖，为的是当两个分离的对象的内容和类型相配的话，返回真值

**64. 数组有没有length()这个方法? String有没有length()这个方法？**

答：没有这个方法但是有length这个属性。

**65. sleep() 和 wait() 有什么区别?**

答：sleep()方法是使线程停止一段时间的方法。在sleep 时间间隔期满后，线程不一定立即恢复执行。这是因为在那个时刻，其它线程可能正在运行而且没有被调度为放弃执行，除非(a)“醒来”的线程具有更高的优先级

(b)正在运行的线程因为其它原因而阻塞。

wait()是线程交互时，如果线程对一个同步对象x 发出一个wait()调用，该线程会暂停执行，被调对象进入等待状态，直到被唤醒或等待时间到。

**66. short s1 = 1; s1 = s1 + 1;有什么错? short s1 = 1; s1 += 1;有什么错?**

答：short s1 = 1; s1 = s1 + 1;有错，s1是short型，s1+1是int型,不能显式转化为short型。可修改为s1 =(short)(s1 + 1) 。short s1 = 1; s1 += 1正确。

**67.谈谈final, finally, finalize的区别。**

答：

final—修饰符（关键字）如果一个类被声明为final，意味着它不能再派生出新的子类，不能作为父类被继承。因此 一个类不能既被声明为 abstract的，又被声明为final的。将变量或方法声明为final，可以保证它们在使用中 不被改变。被声明为final的变量必须在声明时给定初值，而在以后的引用中只能读取，不可修改。被声明为 final的方法也同样只能使用，不能重载

finally—再异常处理时提供 finally 块来执行任何清除操作。如果抛出一个异常，那么相匹配的 catch 子句就会 执行，然后控制就会进入 finally 块（如果有的话）。

finalize—方法名。Java 技术允许使用 finalize() 方法在垃圾收集器将对象从内存中清除出去之前做必要的清理 工作。这个方法是由垃圾收集器在确定这个对象没有被引用时对这个对象调用的。它是在 Object 类中定义的 ，因此所有的类都继承了它。子类覆盖 finalize() 方法以整理系统资源或者执行其他清理工作。finalize() 方法是在垃圾收集器删除对象之前对这个对象调用的。

**68. 如何处理几十万条并发数据？**

答：用存储过程或事务。取得最大标识的时候同时更新..注意主键不是自增量方式这种方法并发的时候是不会有重复主键的..取得最大标识要有一个存储过程来获取.

**69.Session有什么重大BUG，微软提出了什么方法加以解决？**

答：是iis中由于有进程回收机制，系统繁忙的话Session会丢失，可以用Sate server或SQL Server数据库的方式存储Session不过这种方式比较慢，而且无法捕获Session的END事件。

**70.进程和线程的区别？**

答：进程是系统进行资源分配和调度的单位；线程是CPU调度和分派的单位，一个进程可以有多个线程，这些线程共享这个进程的资源。

**71.堆和栈的区别？**

答： 栈：由编译器自动分配、释放。在函数体中定义的变量通常在栈上。

堆：一般由程序员分配释放。用new、malloc等分配内存函数分配得到的就是在堆上。

**72. 成员变量和成员函数前加static的作用？**

答：它们被称为常成员变量和常成员函数，又称为类成员变量和类成员函数。分别用来反映类的状态。比如类成员变量可以用来统计类实例的数量，类成员函数负责这种统计的动作。

**73. 请指出GAC的含义？**

答：全局程序集缓存。

**74.在c#中using和new这两个关键字有什么意义，请写出你所知道的意义？using 指令和语句 new 创建实例 new 隐藏基类中方法。**

答：using 引入名称空间或者使用非托管资源

new 新建实例或者隐藏父类方法

**75.什么是虚函数？什么是抽象函数？**

答：虚函数：没有实现的，可由子类继承并重写的函数。抽象函数：规定其非虚子类必须实现的函数，必须被重写。

**76. C#中 property 与 attribute的区别，他们各有什么用处，这种机制的好处在哪里？**

答：一个是属性，用于存取类的字段，一个是特性，用来标识类，方法等的附加性质

**77. c#中的三元运算符是？**

答：？：

**78.当整数a赋值给一个object对象时，整数a将会被？**

答：装箱。

**79.类成员有\_\_\_\_\_种可访问形式？**

答：this.类成员;

new Class（）类成员

**80. public static const int A=1;这段代码有错误么？是什么？**

答：const不能用static修饰。

**81.float f=-123.567F; int i=(int)f;i的值现在是\_\_\_\_\_?**

答：-123。

**82.委托声明的关键字是\_\_\_\_\_\_?**

答：delegate.

**83.用sealed修饰的类有什么特点？**

答：密封，不能继承。

**84.如何在C#中实现继承？**

在类名后加上一个冒号，再加上基类的名称。

**85.一个进程可以访问的Windows地址空间有多大？等于系统的虚拟内存大小么？这两方面将对系统的设计产生什么样的影响？**

答：4G，虚拟内存大小是看你物理内存大小来定的啊，实际进程在windows上默认是可以访问2G内存，通过一个/3g boot参数，可以用到3G

**86.下面这个命令是做什么的？gacutil /l | find /i “system”**

答：全局缓存工具，gacutil /l列出全局程序集缓存的内容。/i “system”将程序集安装到全局程序集缓存中。

**87.列举ADO.net中的五个主要对象**

答.Connection,Command,DataAdapter,DataSet,DataTable,DataView等

**88.从数据库读取记录，你可能用到的方法有：**

答： ExecuteScalar()

Fill()

ExecuteReader()

**89. 概述反射和序列化**

答：反射:程序集包含模块，而模块包含类型，类型又包含成员。反射则提供了封装程序集、模块和类型的对象。您可以使用反射动态地创建类型的实例，将类型绑定到现有对象，或从现有对象中获取类型。然后，可以调用类型的方法或访问其字段和属性

序列化:序列化是将对象转换为容易传输的格式的过程。例如，可以序列化一个对象，然后使用 HTTP 通过 Internet 在客户端和服务器之间传输该对象。在另一端，反序列化将从该流重新构造对象。

**90.概述o/r mapping 的原理**

答：利用反射，配置 将类于数据库表映射

**91.用.net做B/S结构的系统，您是用几层结构来开发，每一层之间的关系以及为什么要这样分层？**

答：a) 一般为3层

b) 数据访问层，业务层，表示层。

c) 数据访问层对数据库进行增删查改。

d) 业务层一般分为二层，业务表观层实现与表示层的沟通，业务规则层实现用户密码的安全等。

e) 表示层为了与用户交互例如用户添加表单。

f) 优点：分工明确，条理清晰，易于调试，而且具有可扩展性。

g) 缺点：增加成本。

**92. Session有什么重大BUG，有什么方法加以解决？**

答：是iis中由于有进程回收机制，系统繁忙的话Session会丢失，可以用Sate server或SQL Server数据库的方式存储Session不过这种方式比较慢，而且无法捕获Session的END事件。

**93.每次我们请求一个asp.net页面时，我们都经历着同样的过程,我们称它为asp.net页面生命周期！每个页面的生命周期为用户的每一次访问，也就是说每一次客户端与服务器之间的一个往返过程.请你说一说这个过程它们都发生了那些事件？**

答:(1). Page\_Init();

(2). Load ViewState and Postback data;

(3). Page\_Load();

(4). Handle control events;

(5). Page\_PreRender();

(6). Page\_Render();

(7). Unload event;

(8). Dispose method called;

**94. 什么是ASP.net中的用户控件？(用户控件和自定义控件有什么区别?)**

答:主要是分清这两个区别，

(一).用户控件和自定义控件概述

用户控件(UserControl): 扩展名为\*.ascx,跟\*.aspx在结构上相似，是指页面中

加载的功能块,只是用户控件不能单独作为页面运行,必须嵌入到\*.aspx页面或其它用

户控件中使用.

自定义控件，跟HtmlControl或WebControl相似，编译后可以添加引用到工具栏里

面，直接用鼠标拖动使用.

**95..net中读写数据库需要用到那些类？他们的作用？**

答：DataSet:数据存储器。

DataCommand:执行语句命令。

DataAdapter:数据的集合，用语填充。

**96.ASP.net的身份验证方式有哪些？分别是什么原理？**

答：（1）Windows 身份验证提供程序

提供有关如何将 Windows 身份验证与 Microsoft Internet 信息服务 (IIS) 身份验证结合使用来确保 ASP.NET 应用程序安全的信息。

（2）Forms 身份验证提供程序

提供有关如何使用您自己的代码创建应用程序特定的登录窗体并执行身份验证的信息。使用 Forms 身份验证的一种简便方法是使用 ASP.NET 成员资格和 ASP.NET 登录控件，它们一起提供了一种只需少量或无需代码就可以收集、验证和管理用户凭据的方法。

（3）Passport 身份验证提供程序

提供有关由 Microsoft 提供的集中身份验证服务的信息，该服务为成员站点提供单一登录和核心配置

**97.在.net中，配件的意思是？**

答：程序集。（中间语言，源数据，资源，装配清单）

**98. 常用的调用WebService的方法有哪些？**

答：1.使用WSDL.exe命令行工具。

2.使用VS.NET中的Add Web Reference菜单选项

**99. net Remoting 的工作原理是什么？**

答：服务器端向客户端发送一个进程编号，一个程序域编号，以确定对象的位置。

**100.简要谈一下您对微软.NET 构架下remoting和webservice两项技术的理解以及实际中的应用。**

答：WS主要是可利用HTTP，穿透防火墙。而Remoting可以利用TCP/IP，二进制传送提高效率。

**101.公司要求开发一个继承System.Windows.Forms.ListView类的组件，要求达到以下的特殊功能：点击ListView各列列头时，能按照点击列的每行值进行重排视图中的所有行 (排序的方式如DataGrid相似)。根据您的知识，请简要谈一下您的思路**

答：根据点击的列头,包该列的ID取出,按照该ID排序后,在给绑定到ListView中。

**102. ASP。NET与ASP相比，主要有哪些进步？**

答：asp解释形，aspx编译型，性能提高，有利于保护源码。

**103.向服务器发送请求有几种方式？**

答：get,post

**104. .DataReader与Dataset有什么区别？**

答：一个是只能向前的只读游标，一个是内存中的表。

**105. 什么叫做SQL注入，如何防止？请举例说明。**

答：利用sql关键字对网站进行攻击。过滤关键字'等

**106. 什么是反射？**

答：动态获取程序集信息

**107. Singleton如何写设计模式**

答：static属性里面new ,构造函数private

**108.什么是Application Pool？**

答：Web应用，类似Thread Pool，提高并发性能。

**109.什么是XML？**

答：XML即可扩展标记语言。eXtensible Markup Language.标记是指计算机所能理解的信息符号，通过此种标记，计算机之间可以处理包含各种信息的文章等。如何定义这些标记，即可以选择国际通用的标记语言，比如HTML，也可以使用象XML这样由相关人士自由决定的标记语言，这就是语言的可扩展性。XML是从SGML中简化修改出来的。它主要用到的有XML、XSL和XPath等。

**110. XML 与 HTML 的主要区别**

答：1. XML是区分大小写字母的，HTML不区分。

2. 在HTML中，如果上下文清楚地显示出段落或者列表键在何处结尾，那么你可以省略</p>或者</li>之类的结束 标记。在XML中，绝对不能省略掉结束标记。

3. 在XML中，拥有单个标记而没有匹配的结束标记的元素必须用一个 / 字符作为结尾。这样分析器就知道不用 查找结束标记了。

4. 在XML中，属性值必须分装在引号中。在HTML中，引号是可用可不用的。

5. 在HTML中，可以拥有不带值的属性名。在XML中，所有的属性都必须带有相应的值。

**111 .net的错误处理机制是什么？**

答：.net错误处理机制采用try->catch->finally结构，发生错误时，层层上抛，直到找到匹配的Catch为止。

**112.C#中的表单界面上，有一个DataGrid控件，如何将SQL数据库里的一个表中的数据显示在这个控件上，请描述一下操作方法及步骤：**

答：(1) 根据连接字符串，创建数据库连接对象，并根据连接对象创建 Command对象。

(2) 根据SQL语句，查询表信息。

(3) 使用DataApapter对象，将查询结果填充到DataSet中的objDataset数据表里。

(4) 将DataGrid的数据源设置成objDataset数据表，实现数据绑定。

**113. 在C#中，有几种连接到数据库的方式，请列举出并分别说明它们可以连接哪些数据库**

（1）Sql.Net Framework 连接sql server7.0或以上更高版本数据库。

（2）Oledb.Net Framework连接Access和sql server6.5或以下版本数据库。

（3）Odbc.Net Frameworka(桥连)可以连接sql server数据库和odbc公开数据源。

（4）Oracle.Net Framework连接Oracle数据库。

**114.请描述一下在ASP.NET中，Application对象、Session对象、Cookies对象的用法及不同之处，及它们与传统的ASP有没有不同之处？**

答：Application对象是作用于整个程序运行期的状态对象。

Session对象是会话期状态保持对象，用于跟踪单一用户的会话。

Cookies对象客户端保持会话信息的一种方式。

**115. 试简单描述以下扩展名程序或系统档案的角色**

答: .ascx , .aspx , .dll , .vb , .xml, css , js, global.asax ,web.config

Ascx后缀表示用户控件

Aspx后缀表示页面

.dll表示类库集

.vb是vb.Net的源文件

.xml表示是用可扩展标记语言编写的文件

Css表示样式表文件

Js表示javascript脚本文件

Global.asax文件在应用程序的根目录中，负责处理Application\_Start,Application\_End,Session\_Start,Session\_End事件

Web.Config文件存储应用程序的配置信息

**116..net中读写XML的类都归属于哪些命名空间？**

答:System.Xml

**117.相对于.NET 1.1，.NET2.0为了避免过度的boxing/unboxing造成的系统开销提供了哪些支持？**

**118.请谈谈面向对象的特性及你对面向对象的理解？**

**119．简述Javascript中的$符号表示什么？**

**120．列举ADO.NET中的五大主要对象及其作用？**

**121.简述接口和抽象类的异同？你选择使用接口和抽象类的依据是什么？**

**122.请列举ADO.NET中读取数据的方法及其工作原理？**

**123. 简述 private、 protected、 public、 internal 修饰符的访问权限。**

**124 .列举ASP.NET 页面之间传递值的几种方式。**

**125. C#中 property 与 attribute的区别，他们各有什么用处，这种机制的好处在哪里？**

**126. 讲一讲你理解的web service,在dot net framework中，怎么很好的结合xml?（讲概念就行了）**

**127. C#, Java 和 c++的特点，有什么相同的地方，不同的地方，C#分别从c++和java中吸取了他们那些优点？**

**128. C#可否对内存进行直接的操作？**

**129. 用Visual C++ 6.0编写的代码(unmanaged code)，如何在CLR下和其他dot net component结合？**

**130. 以前作过的一些项目？介绍一下自己的编程经验。(这是蒙混不过去的，基本一问就知道是真是假)**

**131： 维护数据库的完整性、一致性、你喜欢用触发器还是自写业务逻辑？为什么**

**132：ADO.NET相对于ADO等主要有什么改进？**

**133：ASP.NET与ASP相比，主要有哪些进步？**

**134：C#中的委托是什么？事件是不是一种委托？**

**135：描述一下C#中索引器的实现过程，是否只能根据数字进行索引？**

**136：配置异常(Debug | Exceptions) 每个异常都可以配置。**

**137.健壮程序的处理**

1) throw 对应的异常，不能够慎用Catch(Exception e) 这种捕获方法.

2) 异常转义抛出,给用户以合理的提示,不要抛出假异常.

3) 给出断言Trace.Assert() 便于调试.

4) 在程序的最后一层,最好要处理一下异常,但是同时要记录日志Log,既要保证用户没有中断操作,又要给自己的改进带来依据(把异常的

日志返回 到开发人员手中),如果有Reporting Server 更好.

5) 总之,不要轻易吃掉异常,只能增加开发的成本

**138.类的修饰符**

1) internal 项目内部用

2) public 默认情况

3) abstract 抽象的,不可以实例化,只能够继承,可有抽象成员

4) sealed 密封的,不能够继承,可以是公有的,也可以是内部的.

**139.所有的接口成员都必须在支持该接口的类中执行,如果不想使用给定的接口成员,可以提供一个"空" 的执行方法(没有函数代码)**

**140.基类必须是继承列表中的第一项.**

**141.GetHashCode 用以对象的散列函数,返回一个以压缩形式标识的对象的值.**

**142.抽象类和接口的区分:**

1) 派生类只能够继承一个抽象类,但可以使用任意多接口.

2)抽象类可有抽象成员(但必须在派生类中执行,否则派生类本身必须是抽象的),接口成员必须在使用接口的类上执行-没有代码体接口成员

被定义为公有的,但是抽象类的成员也可以是私有的.接口不包括静态成员,字段,变量.

3)抽象类主要用于对象系列的基类,共享某些主要特性,例如共同的目的和结构,接口主要是供类使用,这些类在基础水平上有所不同,

但仍然可以完成某些相同的任务.

**143.成员复制简单的对象可以通过System.Object 的MemberwiseClone() 方法完成,称为浅度复制;因为他没有考虑引用类型的成员.执行一个接口**

IConeable 必须执行Clone()的方法,深度复制.

**144.virtual 方法可以重写,abstract 方法必须重写,extern 方法定义放在其他地方.如果使用了override,也可以使用sealed指定在派生类中不能对这个方法做出进一步修改.**

**145.隐藏基类的实现：空方法体；new public void DoSomething()**

**146.IEnumerable 实现了循环集合中项的功能.**

ICollection 继承IEnumerable 获取集合中项的个数,并把项复制到一个数组中.

IList 继承IEnumerable 和 ICollection 提供集合项列表，并可以访问这些项

IDictionray 继承IEnumerable 和 ICollection类似于IList ，但提供了通过键码值而不是索引访问的项列表.

CollectionBase的接口有IList,ICollection,IEnumerable

Array

**147.IDictionary 简化的DictionaryBase,要遍历内容其Item 为： DictionaryEntry结构**

**148.转换运算符中关键字: implicit,explicit.**

**149.is 运算符不检查两个类型是否相同,但可以检查类型是否兼容,如果转换成功,则返回true.**

**150.as 运算符转换要么成功,否则就返回null.**

**151.定制异常,处理自定义异常.**

**152.事件处理.**

**153.控件属性设置，例如：设定　[Conditional("DEBUG")] call some method assigned by the attribute only available if DEBUG is defined......**

154.调用堆栈，断点，命令窗口，输出，自动窗口，局部变量，监视

Immediate 模式：调试和计算表达式

Command 模式：发布命令

计算表达式：在表达式前加上问号

Command -〉Immediate 输入immed 不需要大于号，否则输入〉cmd 在Immediate 模式下临时进入Command 模式

（执行单条命令）输入〉alias，更多命令查看MSDN

局部变量，察看断点前后三行的值

**155.描述下C#中的程序集？程序集是有哪两部分组成？**

答：程序集是自我描述的安装单元，由一个或多个文件组成，一个程序集可以是一个包含元数据的DLL或EXE，也可以由多个文件组成，例如资源文件、元数据、DLL和EXE。程序集的安装仅是复制所有的文件而已，使用xcopy命令即可完成安装，有两种类型：私有的和共有的。

**156.请写出 BOOL flag 与“零值”比较的 if 语句？请写出 char p 与“零值”比较的 if 语句？**

答：if(flag){};if(!flag)。

**157.说出下面几个函数的区别:**   
private void test(string str){…} ：通过值传送变量  
private void test(ref string str){…} ：值参数通过引用传递给方法，必须先初始化str  
private void test(out string str){…}：str通过引用被传送，从test中返回时，保留该方法对其的改变

**158.什么是强类型系统？**

答：C# 是强类型语言；因此每个变量和对象都必须具有声明类型

1. **值类型和引用类型的区别？**

答：值类型直接存储其值，引用类型存储对值的引用。值类型存储在堆栈中，而引用类型存储在托管堆上。值类型包括：简单类型、结构类型、枚举类型；引用类型包括：类、数组、接口、委托、字符串、数组。

1. **如何理解委托？**

答：有点类似于指针，但他是类型安全的，通过对于方法特征和返回类型的声明，封装了具有相同特征和返回类型的方法。不关心该方法是什么类型的对象，甚至是静态的还是实例的。委托可以连接在一起，可以对一个事件调用多个方法。

161． **事务是什么？**

答：在许多大型、关键的应用程序中，计算机每秒钟都在执行大量的任务。更为经常的不是这些任务本身，而是将这些任务结合在一起完成一个业务要求，称为事务。当某一个任务失败时，就恢复到指定的恢复点，这个叫回滚。

162． **解释重载函数、虚拟函数的概念？**

答：重载方法：是指使用相同的方法名，但指定不同的参数（个数，类型）。

虚拟方法：在基类中对要实现多态性的方法加上virtual关键字，在派生类中的同名方法前面，加上override关键字修饰，并改写该方法的代码。

**163．请简述以下两个for循环的优缺点**

|  |  |
| --- | --- |
| for (i=0; i<N; i++)  {  if (condition)  DoSomething();  else  DoOtherthing();  } | if (condition)  {  for (i=0; i<N; i++)  DoSomething();  }  else  {  for (i=0; i<N; i++)  DoOtherthing();  } |

答：其实这两个for循环在不同的环境下不同。总的来说前面的代码简洁，但是多执行了N－1次，效率底些，后面的效率高些，但是不够简洁。

**164．描述怎样区分ExcuteNonQuery ，ExcuteDataReader和ExcuteScalar方法？**

答：ExcuteNonQuery返回受影响的行数；ExcuteDataReader返回DataReader；ExcuteScalar返回查询结果的第一行的一列的值，忽略其他行列。

**165. 列出ADO。NET中读写数据库的主要的几个类？它们的作用？**

答：Connection数据库连接；Command执行SQL语句或存储过程；DataAdapter用于填充DataSet和更新数据库的一组数据命令和数据库连接，断开模式；DataReader，读取数据库信息，是只读向前的；DataSet数据在内存中的缓存。

**166. 什么是中间语言（IL）？它的作用？**

答：代码在执行前都要被编译为托管代码，与CPU无关的指令集。

其作用：支持平台无关性，还支持语言的互操作性。

**167.触发器可以强制限制，这些限制比用 CHECK 约束所定义的更复杂。与 CHECK 约束不同的是，触发器可以引用其它表中的列。**

**168．在ASP.NET中页面之间传递值的方式？(尽可能叙述)**

答： session(viewstate) 简单，但易丢失

application 全局

cookie 简单，但可能不支持，可能被伪造

input ttype="hidden" 简单，可能被伪造

url参数 简单，显示于地址栏，长度有限

数据库 稳定，安全，但性能相对弱

**169．分析类和结构的异同？**

答：**类**是引用类型，可以继承类、接口和被继承，有默认的构造函数，有析构函数，可以使用abstract和sealed，有protected修饰符，必须使用new初始化。

**结构**是值类型，只能继承接口，不能被继承，没有默认的构造函数，可以创建，没有析构函数，不可以用abstract和sealed，没有protected修饰符，可以不用new初始化。  
**如何选择使用结构还是类：**   
1．堆栈的空间有限，对于大量的逻辑的对象，创建类要比创建结构好一些   
2．结构表示如点、矩形和颜色这样的轻量对象，例如，如果声明一个含有 1000 个点对象的数组，则将为引用每个对象分配附加的内存。在此情况下，结构的成本较低。   
3．在表现抽象和多级别的对象层次时，类是最好的选择   
4．大多数情况下该类型只是一些数据时，结构是最佳的选择

**171．叙述ASP。NET中的equiredFieldValidator和RegularExpressionValitor验证控件及作用？**

答：equiredFieldValidator是必须字段验证，RegularExpressionValitor是正则表达式验证。

**172．叙述下Using 的用法？**

答：1. using指令。using + 命名空间名字，这样可以在程序中直接用命令空间中的类型，而不必指定类型的详细命名空间。

2.using别名。using + 别名 = 包括详细命名空间信息的具体的类型。当一个cs引用多个命名空间时，而有相同的类型时可以这样做。比如namespace1 和namespace2下面都有myClass类时，可以using class1 ＝ namespace1.myClass;using class2 = namespace2.myClass。

3.using语句，定义一个范围，在范围结束时处理对象。

**173．C#中接口和类有什么异同？**

答：接口包括属性、方法、事件，是负责功能的声明和定义的，功能的实现从继承他的类中实现。类是实现具体功能的，可以继承多个接口，但是只能继承一个基类。

**174．什么是装箱和拆箱？**

答：装箱就是把值类型转换成引用类型，拆箱是把引用类型转换成值类型。

**175．什么是WEBSERVICE ？**

答：web服务，是利用SOAP（简单对象访问协议，Simple Object Access Protocol）在HTTP上执行远程方法调用的一种方法，也可以使用WSDL（Web Service Description Language，Web服务描述语言）来完整的描述Web服务，基于XML标准。

**176．软件开发过程一般有几个阶段？**

答：分析，开发，测试，维护

177．在ASP。NET中自定义控件是什么？

答：自定义控件，跟HtmlControl或WebControl相似，编译后可以添加引用到工具栏里面，直接用鼠标拖动使用

178．ASP.NET的Application、Session、Cookie、ViewState和Cache等变量的区别是什么？

答：Application：能够包含任意大小的信息，在整个应用程序周期中有效，是面对所有应用程序用户的，保存于服务器端。

1. Session：能够保存相对少量的、简单的数据，一般在用户活动时间加上一端Timeout设定时间中有效（通常是20分钟Timeout），这是面向单个用户的信息，保存于服务器端。
2. Cookie：保存少量的、简单的数据，一般不允许超过4KB，使用期限可以进行灵活设定，面向单个用户，保存于浏览器中（也就是客户端）。
3. ViewState：保存少量的数据，在一个Web页面周期内有效，面向单个用户，保存于客户端。
4. Cache：缓存，大小任意，可以灵活设定可用时间，一般针对所有用户，保存于客户端。

179．解释下在ASP.NET中Response的作用?

答：Response对象用于向客户端浏览器发送数据，用户可以使用该对象将服务器的数据以HTML的格式发送到用户端的浏览器，它与Request组成了一对接收、发送数据的对象，这也是实现动态的基础。

180．解释下在ASP.NET中Request的作用?

答：Request对象的主要作用就是：在服务器端接受并得到从客户端浏览器提交或上传的信息。包括从HTML表格用POST方法或GET方法传递的参数、cookie和用户认证。

181. C#中能够使用指针吗?如果不能使用说明理由,如果能使用说出使用方法?

答：可以使用。只能在标记为unsafe关键字的地方使用，任何方法、类、结构、成员都能标记为unsafe，也可以把方法中的一个代码块标记为unsafe，但要注意，不能把局部变量本身标记为unsafe，除非在unsafe的方法或代码块中声明。

182．什么是中间语言（IL）？它的作用？

答：代码在执行前都要被编译为托管代码，与CPU无关的指令集。

其作用：支持平台无关性，还支持语言的互操作性。

183．什么是装箱和拆箱？

答：值类型转换成引用类型是装箱，引用类型转换成值类型是拆箱。

184.出ADO。NET中读写数据库的主要的几个类？它们的作用？

答：Connection数据库连接；Command执行SQL语句或存储过程；DataAdapter用于填充DataSet和更新数据库的一组数据命令和数据库连接，断开模式；DataReader，读取数据库信息，是只读向前的；DataSet数据在内存中的缓存。

185.C#中接口和类有什么异同？

答：**类**是引用类型，可以继承类、接口和被继承，有默认的构造函数，有析构函数，可以使用abstract和sealed，有protected修饰符，必须使用new初始化。

**结构**是值类型，只能继承接口，不能被继承，没有默认的构造函数，可以创建，没有析构函数，不可以用abstract和sealed，没有protected修饰符，可以不用new初始化。  
**如何选择使用结构还是类：**   
1．堆栈的空间有限，对于大量的逻辑的对象，创建类要比创建结构好一些   
2．结构表示如点、矩形和颜色这样的轻量对象，例如，如果声明一个含有 1000 个点对象的数组，则将为引用每个对象分配附加的内存。在此情况下，结构的成本较低。   
3．在表现抽象和多级别的对象层次时，类是最好的选择   
4．大多数情况下该类型只是一些数据时，结构是最佳的选择

186.解释ASP。NET中的code-behind?

答：代码隐藏，显示逻辑的代码和处理逻辑的代码分离

187.什么是ASP。NET中的用户控件？

答：是asp.net代码创建的控件，就像在标准的ASP.NET Web页面中创建控件一样，可以在多个ASP.NET页面中重用它们。

188.在C＃中，string str = null 与 string str = “” 请尽量说明其中的区别。

答：string str＝null没有分配空间，string str＝“”分配了空间

189. 什么是受管制代码？

答：由公共语言运行环境进行管理的代码，它主要负责内存管理。在.net平台上运行的代码都是受管制代码

190．C#中常用的访问形式？它们各自的作用域？

答：public 不受限制的访问。

private 只能在本类中访问

protected 本类和继承他的类中访问

internel表示在同一个应用程序（application）或类库（library）中都能使用

191.CTS，CLS，CLR分别作何解释？

答： CTS:公共类型系统.主要是为了语言之间的进行标准化处理.

CLS:公共语言规范.主要是确保代码可以在任何语言中访问的最小标准集体

CLR:公共语言运行时.主要是管理代码..处理,加载代码,以及所有服务的代码

192. ASP。NET和ASP的主要优势？

答：ASP.Net的优势：

ASP.Net摆脱了以前ASP使用脚本语言来编程的缺点，理论上可以使用任何编程语言包括C++ , VB , JS等等，当然，最合适的编程语言还是MS为.Net Frmaework专门推出的C(读c sharp)，它可以看作是VC和Java的混合体吧，尽管MS自己讲C#内核中更多的象VC，但实际上我还是认为它和Java更象一些吧。首先它是面向对象的编程语言，而不是一种脚本，所以它具有面向对象编程语言的一切特性，比如封装性、继承性、多态性等等，这就解决了刚才谈到的ASP的那些弱点。封装性使得代码逻辑清晰，易于管理，并且应用到ASP.Net上就可以使业务逻辑和Html页面分离，这样无论页面原型如何改变，业务逻辑代码都不必做任何改动；继承性和多态性使得代码的可重用性大大提高，你可以通过继承已有的对象最大限度保护你以前的投资。并且C#和C++、Java一样提供了完善的调试/纠错体系。

193. NET的错误处理机制是什么？

答：try{可能要出错的代码}

catch{扑捉到错误后的处理}

finally{不论代码是否出错都要执行}

194. 解释Page.IsPostBack属性？

答：使用者第一次浏览这个网页时Page.IsPostBack 会传回False,不是第一次浏览这个网页时就传回True

195．.net framework包含哪几种技术？

答：Application Framework,Domain Framework

196．解释下C#中的方法的重载？

答：重载方法：是指使用相同的方法名，但指定不同的参数（个数，类型）。

197．事务是什么？

答：在许多大型、关键的应用程序中，计算机每秒钟都在执行大量的任务。更为经常的不是这些任务本身，而是将这些任务结合在一起完成一个业务要求，称为事务。当某一个任务失败时，就恢复到指定的恢复点，这个叫回滚。

198．描述下ASP。NET中使用DataReader对象 和DataSet对象的区别？

答：dataset是保存数据的数据结构，断开模式，而DataReader不承担保存数据的责任，需要自己手动关闭连接数据库，它只负责从数据源读取数据到本地而已，是只读向前的，它不是数据结构，而是网络通讯组件的高层封装。

199．分析类和结构的异同？

答：类是引用类型，可以继承类、接口和被继承，有默认的构造函数，有析构函数，可以使用abstract和sealed，有protected修饰符，必须使用new初始化。

结构是值类型，只能继承接口，不能被继承，没有默认的构造函数，可以创建，没有析构函数，不可以用abstract和sealed，没有protected修饰符，可以不用new初始化。

如何选择使用结构还是类：

1．堆栈的空间有限，对于大量的逻辑的对象，创建类要比创建结构好一些

2．结构表示如点、矩形和颜色这样的轻量对象，例如，如果声明一个含有 1000 个点对象的数组，则将为引用每个对象分配附加的内存。在此情况下，结构的成本较低。

3．在表现抽象和多级别的对象层次时，类是最好的选择

4．大多数情况下该类型只是一些数据时，结构是最佳的选择

200．解释ADO。NET中的使用Connection 类的CLOSE（）和DISPOSE（）的区别？

答：close只是关闭，dispose是释放资源。

201．叙述ADO。NET中的DataAdapte类的作用？

答：断开模式连接数据库，并且填充DataSet。

202．什么是WEBSERVICE ？

答：web服务，是利用SOAP（简单对象访问协议，Simple Object Access Protocol）在HTTP上执行远程方法调用的一种方法，也可以使用WSDL（Web Service Description Language，Web服务描述语言）来完整的描述Web服务，基于XML标准。

203．叙述ASP。NET中的几种验证控件(equiredFieldValidato, CompareValidator, RangeValidator, RegularExpressionValitor, CustomValidator, ValidationSummary)及作用？

答：

* 1. RequiredFieldValidator（必须字段验证）；
  2. CompareValidator（比较验证）；
  3. RangeValidator （范围验证）；
  4. RegularExpresionValidator（正则表达式）；
  5. ValidationSummary（验证总结）；
  6. CustomValidator（自定义验证）

204. 用sealed修饰的类有什么特点

sealed 修饰符用于防止从所修饰的类派生出其它类。如果一个密封类被指定为其他类的基类，则会发生编译时错误。密封类不能同时为抽象类。

sealed 修饰符主要用于防止非有意的派生，但是它还能促使某些运行时优化。具体说来，由于密封类永远不会有任何派生类，所以对密封类的实例的虚拟函数成员的调用可以转换为非虚拟调用来处理。