**DOTNET:**

**1: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？**

**尽可能用约束(包括CHECK、主键、唯一键、外键、非空字段)实现，这种方式的效率最好；其次用触发器，这种方式可以保证何种业务系统访问数据库都能维持数据库的完整性、一致性；最后再考虑用自写业务逻辑实现，但这种方式效率最低、编程最复杂，当为下下之策.**

**2:ADO.NET相对于ADO等主要有什么改进？**

**ADO数据以Recordset形式存储ADO.NET以DataSet形式存储**

**Recordset对数据库持续连接访问ADO.NET提供对数据库断开连接**

**ADO.NET与ADO相比，优势在于提供了数据集和数据适配器，有利于实现分布式处理，降低对数据库服务器资源的消耗**

**3:ASP.NET与ASP相比，主要有哪些进步？**

**asp.net？可以使用强类型语言**

**页面是编译，执行速度快，增加安全性和可靠性，通过继承机制来支持代码的重用，提供声明性服务器控件，减少代码行数**

**Asp需要解释，执行速度慢，重用代码不方便，没有调试机制**

**4:C#中的委托是什么？事件是不是一种委托?**

**委托可以把一个方法作为参数代入另一个方法**

**委托可以理解为指向一个函数的引用**

**是，是一个特殊的委托**

**5：new有几种用法**

**第一种：new Class();**

**第二种：覆盖方法**

**public new XXXX(){}**

**第三种：new约束指定泛型类声明中的任何类型参数都必须有公共的无参数构造函数**

**6:概述反射和序列化**

**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。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**

**7:类成员有几种访问形式**

**可访问性：public、protected、private、internal**

**8:概述.NET里对remoting和webservice两项技术的理解和实际中的应用。**

**远程逻辑调用，remoing接口只能用在.net中**

**9:用.net做B/S结构的系统，您用几层结构来开发，每一层的关系以及为什么要这样分层？**

**一般为3层**

**数据访问层，业务层，表示层**

**数据访问层：对数据库进行增删改查**

**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：为了与用户交例**

**例如：用户添加表单。.**

**优点：分工明确，条理清晰，易于调试，具有可扩展性。**

**缺点：增加成本。**

**10:什么是装箱和拆箱？**

**装箱：值类型到引用类型的装换**

**拆箱：引用类型到值类型的装换**

**11:什么是方法重载和重写？**

**方法重载：方法名相同，不同个数的参数 ，不同参数类型，方法重载与返回值没有任何关系**

**重写：方法名、参数列表、返回值一样，提供不同的实现,子类继承父类**

**12:CTS、CLS和CLR分别作何解释**

**CTS：通用语言系统**

**CLS：通用语言规范**

**CLR：公共语言运行库**

**13:ADO.NET中五大核心对象有哪些？分别描述一下。**

**Connection：对象用于在应用程序和数据库之间的连接**

**Command：可以检索和操纵数据库中的数据**

**DataAdapter：对象充当DataSet和数据源之间用于检索和保存的桥接器**

**DataSet：对象从数据库检索的数据可以存储在其中能够以XML形式保存**

**DataReader：是查询结果的一种只进，只读的视图。不具有DataSet的任何复杂功能 。所以可用加快访问和查看数据的速度 。不提供断开式访问。**

**14:C#中的委托是什么？事件是不是一种委托？**

**委托可以把一个方法作为参数代入另一个方法**

**委托可以理解为指向一个函数的引用**

**是，是一个特殊的委托**

**15：进程线程分别怎么理解?**

**一个应用程序有一个或者多个进程，一个进程由多个线程组成**

**16:如何理解.net中的垃圾回收机制。**

**GC是垃圾收集器，程序员不用担心内存合理，因为垃圾收集器会自动进行处理要求垃圾改集。**

**17:列举ASP.NET页面之间传递值的几种方式。（ASP.NET对象）**

**QueryString、Session、Server.Transfer**

**18:请编程实现一个冒泡排序算法？**

**int[] array=new int[\*];**

**int temp=0;**

**for(int i=0;i<array.Length-1;i++)**

**{**

**for(int j=i+1;j<array.Length;j++)**

**{**

**if(array[j]<array[i])**

**{**

**temp=array[i];**

**array[i]=array[j];**

**array[j]=temp;**

**}}}**

**19:简要谈一下你对微软.NET构架下remoting和webservice两项技术的理解和应用**

**webservice主要是可利用HTTP穿透防火墙。**

**Remoting可以利用TCP/IP,二进制传送提高效率**

**20:请说明.net中的错误处理机制，并举例。**

**异常是从Exception类继承的对象。**

**异常是从发生问题的代码区域引发，然后沿堆栈向上传递，直到应用程序处理它或程序终止。**

**Try{**

**//执行代码，不确定是否会出错**

**}**

**Catch{**

**//出错处理**

**}**

**Finally{**

**//无论怎样，都要执行**

**}**

**21:SQL service中，向表中插入了新数据,如何快捷的得到自增字段的当前值。**

**insert into 表名(,,) values(‘’,,)**

**22:请解释ASP.NET中什么方式进行数据验证？**

**.net中提供了几个数据验证控件，可以在服务器端或者客户端进行验证。**

**23:ASP.NET中共有几种类型的控件？各有什么区别？**

**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**

**24:WEB控件及HTML服务端控件能否调用客户端方法？如果能，请解释如何调用？**

**能，服务器端控件在html中表现形式还是html标记，所以可以执行客户端**

**有多种方式：**

**(1)Control.Attributes["onclick"]="….";**

**(2)<script for=”controlName” event="onclick"></script>**

**25:请解释ASP.NET中的web页面与其隐藏类之间的关系？**

**继承的关系:其实页面与其隐藏类之间就是一个部分类的关系**

**这样做是为了把展现与处理逻辑分开**

**26:ASP.NET页面生命周期?**

**页面初始化-->页面被加载-->验证-->服务器控件进行验证-->控件事件-->页面显示-->页面卸载**

**27:值类型和引用类型的区别？**

**结构是值类型，类是引用类型，所以传结构就是值类型的应用，传对象或类就是引用类型**

**28:AJAX工作原理？**

**Ajax就是异步提交请求，实现页面的局部刷新，核心对象是XmlHttpRequest,**

**相关技术有：javascript、Dom、XHTML、CSS**

**29:C#中接口和类有什么异同？**

**接口只能包含抽象方法，不能包含任何方法实现，不能创建接口实例，**

**接口成员没有访问修饰符，接口成员必须是方法，属性，事件或者索引器，**

**不能包含常数，字段，运算符，也不能有静态成员。**

**30:面向对象的基本特征？**

抽象：抽象就是忽略一个主题中与当前目标无关的那些方面，以便充分的注意与当前目标有关的方面，抽象包括两个方面，一个是过程抽象，二是数据抽象

封装：封装是把过程和数据包围起来，对数据的访问只能通过已定义的界面，面向对象计算始于这个基本概念，即现实世界可以被描绘成一系列完全自治，封装的对象，这些对象通过一个受保护的接口访问其他对象

继承：继承是一种联结类的层次模型，并且允许和鼓励类的重用，它提供了一种明确表达共性的方法，对象的一个新类可以从现在的类中派生，这个过程成为继承，新类继承了原始类的特性，新类成为原始类的派生类，而原始类称为新类的基类，派生类可以从他的基类那里继承方法和实例变量，并且类可以修改或增加新的方法使之更加适合特殊的需求，

多态性：多态性是指允许不同类的对象对同一消息作出响应，多态性包括参数化多态性和包含多态性，多态性语言具有灵活，抽象，行为共享，代码共享的优势，很好的解决了应用程序函数同名的问题

**31:ASP.NET MVC各部分的职责和执行过程？**

**应用程序启动，注册路由：**

**当请求到来时，寻找路由表找到负责处理的控制器和动作方法，交给该动作方法处理；动作方法处理完毕，根据寻址规则找到视图，有视图负责页面呈现。**

**32:控制器传递数据到视图有几种方式?**

**(a)ViewData:需要做类型转换。**

**(b)ViewBag:属于动态类型，存取数据不做编译时检查，它和ViewData保存的数据想通。**

**(c)Model:强类型视图，属于强类型数据，编译时检查类型。**

**(d)TempData:数据使用后即消失，可以跨请求传递数据。**

**33:类和结构的区别？**

**结构是值类型**

**结构不支持继承**

**结构不能定义默认的构造函数**

**结构不能定义析构函数**

**结构不能使用初始值设置域值**

**34:动作方法的返回类型可以是值类型吗？为什么？**

**可以，因为框架会自动把值类型封装成ContentResult类型。**

**35:写出一条SQL分页语句：取出表A中31到40记录，ID作为主键.**

**select top 10 \* from A where id not in (select top 30 id from A)**

**36:什么是SQL注入攻击?如何防范?**

**利用sql关键字对网站进行攻击，过滤关键字等**

**所谓SQL注入,就是利用程序员对用户输入数据的合法性检测不严或不检测的特点，故意从客户端提交特殊的代码，从而收集程序及服务器的信息，从而获取想得到的资料。**

**37:Jquery常用选择器种类？**

**基本选择器、层次选择器、表单选择器、过滤选择器**

**38:ORM是什么?**

**对象关系映射，是一种程序技术，用于实现面向对象编程语言里不同类型系统的数据之间的转换。 从效果上说，它其实是创建了一个可在编程语言里使用的--“虚拟对象数据库”。**

**39:使用WEB控件有哪些优势？**

**web控件就是可以在服务器执行的控件**

**优势：回传数据，带有事件驱动**

**40:C/S和B/S模式的区别？**

**B/S：浏览器/服务器模式 ，运行在服务器 部署升级简单方便**

**C/S ：客户端/服务器端模式 ， 运行在客户端 部署升级相对麻烦**

**41:GET和POST的区别？**

**根据HTTP规范，Get用于信息获取，而且应该是安全的和幕的。**

**根据HTTP规范，POST表示可能改变服务器上的资源的请求。**

**42:数据库事务的四大特征？**

**1.原子性：指事务中的操作是一个整体，一起提交一起回滚**

**2.一致性：指事务开始和结束后，表中的数据保持一致状态**

**3.隔离性：指事务之间相互隔离，互不影响**

**4.持久性：指事务结束后对记录的修改是持久的**

**43:string和stringBuilder类有什么区别？**

**String长度是不可变的 StringBuilder是可变的，一般要进行多次修改就用StringBuilder**

**44:const与readonly的区别。**

**Const:值在编译期间确定，readonly在运行时计算出**

**45:存储过程有哪些优势？**

**1、增强了SQL语句的灵活性**

**2、安全性，完整性**

**46:ASP.NET状态保持对象有哪些？**

**Cookie、Session、Application和ViewState**

**47:转发与跳转的区别？**

**转发就是服务器端的跳转A页面提交数据到B页面，B页面进行处理然后从服务器端跳转到其他页面。**

## **跳转就是指客户端的跳转。**

## 请求转发：request.getRequsetDispatcher().forward(requset,response); 重定向：response.Redirect(),

1）转发作用于服务器端，是在服务器内部进行转发。重定向作用于客户端，相当于客户端重新发送一次新的请求。  
2）转发后地址栏不会改变。重定向后地址栏改变。  
3）转发后资源能获取到请求中的数据。重定向后的资源不能获得原请求中的数据。  
4）转发只能在本应用内部资源之间进行。重定向可以跳转到任何网络资源。  
5）转发可以访问受保护资源（WEB-INF里的资源）。重定向不能定位到受保护资源。

request.getRequsetDispatcher().forward(requset,response);

**48:静态类和非静态类的区别？**

**静态类：静态类不能实例化，静态类成员必须是静态的，静态类构造函数无访问修饰符，无参数，只有一个static标志**

**49:三层结构的优势？**

**分工明确，条理清晰，易于调试，具有可扩展性。**

**50：Linq扩展方法有什么要求？**

**扩展方法必须在静态类中定义，是一种特殊的静态方法**

**说明：每一次入厂考试笔试共12道题，其中一道题是人事面试题，2道技术扩展题（近期的真实面试题），余下的从以上50道技术考题中挑选。**