

# .Net 工程师面试笔试宝典



黑马程序员训练营&传智播客.Net 面试笔试宝典是非常珍贵的一套资料，经过黑马程序员训练营&传智播客.Net 就业班学员几年的检验，大部分公司的面试题都是源自面试宝典的。

这套面试题主要目的是帮助那些还没有.Net 软件开发实际工作经验，而正在努力寻找.Net 软件开发工作的朋友在笔试时更好地赢得笔试和面试。由于这套面试题涉及的范围很泛，很广，很杂，大家不可能一天两天就看完和学完这套面试宝典，即使你已经学过了有关的技术，那么至少也需要一个月的时间才能消化和掌握这套面试宝典，所以，大家应该早作准备，从拿到这套面试宝典之日起，就要坚持在每天闲暇之余学习其中几道题目，日积月累，等到出去面试时，一切都水到渠成，面试时就自然会游刃有余了。

回答问题的思路：先正面叙述一些基本的核心知识，然后描述一些特殊的东西，最后再来一些锦上添花的东西。要注意有些不是锦上添花，而是画蛇添足的东西，不要随便写上。把答题像写书一样写。我要回答一个新技术的问题大概思路和步骤是：我们想干什么，干这个遇到了什么问题，现在用什么方式来解决。其实我们讲课也是这样一个思路。

例如，讲 ajax 时，我们希望不改变原来的整个网页，而只是改变网页中的局部内容，例如，用户名校验，级联下拉列表，下拉树状菜单。用传统方式，就是浏览器自己直接向服务器发请求，服务器返回新页面会盖掉老页面，这样就不流畅了。

答题时，先答是什么，再答有什么作用和要注意什么（这部分最重要，展现自己的心得）

答案的段落分别，层次分明，条理清楚都非常重要，从这些表面的东西也可以看出一个人的习惯、办事风格、条理等。

要讲你做出答案的思路过程，或者说你记住答案的思想都写下来。把答题想着是辩论赛。答题就是给别人讲道理、摆事实。答题不局限于什么格式和形式，就是要将自己的学识展现出来！

别因为人家题目本来就模棱两可，你就心里胆怯和没底气了，不敢回答了。你要大胆地指出对方题目很模糊和你的观点，不要把面试官想得有多高，其实他和你就是差不多的，你想想，如果他把你招进去了，你们以后就是同事了，可不是差不多的吗？

关于就业薪水，如果你是应届生，那不能要高工资，要拿高工资，就去中关村！少数人基础确实很好，在校期间确实又做过一些项目，那仍然是可以要到相对高的工资的。

初级程序员薪水：3000-4500

中级程序员薪水：4500-7000

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

高级程序员薪水：7000 以上

公司招聘程序员更看重的是要用到的编码技术、而不是那些业务不太相关的所谓项目经历：

1.公司想招什么样的人 2.公司面试会问什么,3.简历怎么写 4 怎样达到简历上的标准(培训中心教项目的目的)

对于一些公司接到了一些项目，想招聘一些初中级的程序员过来帮助写代码，完成这个项目，你更看重的是他的专业技术功底，还是以前做过几个项目的经历呢？我们先排除掉那些编码技术功底好，又正好做过相似项目的情况，实际上，这种鱼和熊掌兼得的情况并不常见。其实公司很清楚，只要招聘进来的人技术真的很明白，那他什么项目都可以做出来，公司招人不是让你去重复做你以前的项目，而是做一个新项目，业务方面，你只要进了项目团队，自然就能掌握。

所以，大多数招聘单位在招聘那些编码级别的程序员时也没指望能招聘到做过类似项目的人，也不会刻意去找做过类似项目的人，用人单位也不是想把你招进，然后把你以前做过的项目重做一遍，所以，用人单位更看重招进来的人对要用到的编码技术的功底到底怎样，技术扎实不扎实，项目则只要跟着开发团队走，自然就没问题。除非是一些非常专业的行业，要招聘特别高级的开发人员和系统分析师，招聘单位才特别注重他的项目经验和行业经验，要去找行业高手，公司才关心项目和你聊项目的细节，这样的人通常都不是通过常规招聘渠道去招聘进来的，而是通过各种手段挖过来的，这情况不再我今天要讨论的范围中。

技术学得明白不明白，人家几个问题就把你的深浅问出来了，只要问一些具体的技术点，就很容易看出你是真懂还是假懂，很容易看出你的技术深度和实力，所以，技术是来不得半点虚假的，必须扎扎实实。

由于项目的种类繁多，涉及到现实生活中的各行各业，什么五花八门的业务都有，例如，酒店房间预定管理，公司车辆调度管理，学校课程教室管理，超市进销存管理，知识内容管理，等等……成千上万等等，但是，不管是什么项目，采用的无非都是我们学习的那些目前流行和常用的技术。

技术好、经验丰富，则项目做出来的效率高些，程序更稳定和更容易维护些；技术差点，碰碰碰碰最后也能把项目做出来，无非是做的周期长点、返工的次数多点，程序代码写得差些，用的技术笨拙点。如果一个人不是完完全全做过某个项目，他是不太关心该项目的业务的，对其中的一些具体细节更是一窍不知，(如果我招你来做图书管理，你项目经历说你做过汽车调度，那我能问你汽车调度具体怎么回事吗？不会，所以，你很容易蒙混过去的)而一个程序员的整个职业生涯中能实实在在和完完整整做出来的项目没几个，更别说在多个不同行业的项目了，有的程序员更是一辈子都只是在做一个行业的项目，结果他就成了这个行业的专家(专门干一件事的家伙)。

所以，技术面试官通常没正好亲身经历过你简历写的那些项目，他不可能去问你写的那些项目的具体细节，而是只能泛泛地问你这个项目是多少人做的，做了多长时间，开发的过程，你在做项目的过程中有什么心得和收获，用的什么技术等面上的问题，所以，简历上的项目经历可以含有很多水分，很容易作假，技术面试官也无法在项目上甄别你的真伪。

**不是贷款，零学费入学，不 3K 就业不给一分学费**  
**中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.itheima.com)**

简历该怎么写：精通那些技术，有一些什么项目经历

教项目是为了巩固和灵活整合运用技术，增强学习的趣味性，熟悉做项目的流程，或得一些专业课程中无法获得的特有项目经验，增强自己面试的信心。讲的项目应该真实可靠才有价值，否则，表面上是项目，实际上还是知识点的整合，对巩固技术点和增强学习的趣味性，但无法获得实际的项目经验。（项目主要是增加你经验的可信度，获得更多面试机会，真正能不能找到工作，找到好工作，主要看你键盘上的功夫了）

**黑马程序员.Net训练营** <http://net.itheima.com>

# 黑马程序员.Net训练营

**不给1分 钱即可入学！**  
**学习期间 不给1分钱！**  
**不就业 不给1分钱！**

**核心课程：**ASP.NET + MVC + Spring.Net+NHibernate  
WindowsPhone + Html5 + Windows8

**详情请搜索 “黑马程序员.Net训练营”**

**主办：**CSDN、中关村软件园人才中心      **教学实施：**传智播客顶级师资团队

## 技术类面试、笔试题汇总

注：标明\*的问题属于选择性掌握的内容，能掌握更好，没掌握也没关系。

下面的参考解答只是帮助大家理解，不用背，面试题、笔试题千变万化，不要梦想着把题覆盖了，下面的题是供大家查漏补缺用的，真正的把这些题搞懂了，才能“以不变应万变”。回答问题的时候能联系做过项目的例子是最好的，有的问题后面我已经补充联系到项目中的对应的案例了。

### 1、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：当前程序集内可以访问。

### 2、ADO.NET 中的五个主要对象

**Connection**：主要是开启程序和数据库之间的连接。没有利用连接对象将数据库打开，是无法从数据库中取得数据的。Close 和 Dispose 的区别，Close 以后还可以 Open，Dispose 以后则不能再用了。

**Command**：主要可以用来对数据库发出一些指令，例如可以对数据库下达查询、新增、修改、删除数据等指令，以及调用存在数据库中的存储过程等。这个对象是架构在 Connection 对象上，也就是 Command 对象是透过连接到数据源。

**DataAdapter**：主要是在数据源以及 DataSet 之间执行数据传输的工作，它可以透过 Command 对象下达命令后，并将取得的数据放入 DataSet 对象中。这个对象是架构在 Command 对象上，并提供了许多配合 DataSet 使用的功能。

**DataSet**：这个对象可以视为一个暂存区（Cache），可以把从数据库中所查询到的数据保留起来，甚至可以将整个数据库显示出来，**DataSet 是放在内存中的**。DataSet 的能力不只是可以储存多个 Table 而已，还可以透过 DataAdapter 对象取得一些例如主键等的数据表结构，并可以记录数据表间的关联。DataSet 对象可以说是 ADO.NET 中重量级的对象，这个对象架构在 DataAdapter 对象上，本身不具备和数据源沟通的能力；也就是说我们是将 DataAdapter 对象当做 DataSet 对象以及数据源间传输数据的桥梁。**DataSet 包含若干 DataTable、DataTable 包含若干 DataRow**。

**DataReader**：当我们只需要循序的读取数据而不需要其它操作时，可以使用 DataReader 对象。DataReader 对象只是一次一笔向下循序的读取数据源中的数据，这些数据是**存在数据库服务器中的，而不是一次性加载到程序的内存中的，只能（通过游标）读取当前行的数据**，而且这些数据是只读的，并不允许作其它的操作。因为 DataReader 在读取数据的时候限制了每次只读取一笔，而且只能只读，所以使用起来不但节省资源而且效率很好。使用 DataReader 对象除了效率较好之外，因为不用把数据全部传回，故可以降低网络的负载。

ADO.NET 使用 Connection 对象来连接数据库，使用 Command 或 DataAdapter 对象来执行 SQL 语句，并将执行的结果返回给 DataReader 或 DataAdapter，然后再使用取得的 DataReader 或 DataAdapter 对象操作数据结果。



## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

### 3、列举 ASP.NET 页面之间传递值的几种方式。

- 1.使用 QueryString, 如....?id=1; response.Redirect()....
- 2.使用 Session 变量
- 3.使用 Server.Transfer
- 4.Cookie 传值

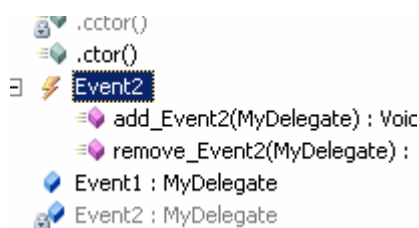
学习 ASP.net 最好的参考资料就是以“不做只会拖控件的菜鸟”为课程宗旨的中关村黑马程序员.Net 训练营 <http://net.ITHeima.com> 的 ASP.Net 课程, 可以下载他们的视频教程自学, 下载地址: <http://edu.csdn.net/heima/aspnetvideo.htm>

### 4、C#中的委托是什么？事件是不是一种委托？事件和委托的关系。

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指针。

委托和事件没有可比性, 因为委托是类型, 事件是对象, 下面说的是委托的对象(用委托方式实现的事件)和(标准的 event 方式实现)事件的区别。事件的内部是用委托实现的。因为对于事件来讲, 外部只能“注册自己+=、注销自己-=”, 外界不可以注销其他的注册者, 外界不可以主动触发事件, 因此如果用 Delegate 就没法进行上面的控制, 因此诞生了事件这种语法。事件是用来阉割委托实例的, 类比用一个自定义类阉割 List。事件只能 add、remove 自己, 不能赋值。事件只能+=、-=, 不能=。加分的补充回答: 事件内部就是一个 private 的委托和 add、remove 两个方法



面试聊: 用 Reflector 查看 .Net 的类的内部实现, 解决问题。

### 5、override 与重载 (overload) 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。重载 (overload) 是面向过程的概念。

Override 是进行基类中函数的重写。Override 是面向对象的概念

### 6、C#中索引器是否只能根据数字进行索引？是否允许多个索引器参数？

参数的个数和类型都是任意的。加分的补充回答: 用 reflector 反编译可以看出, 索引器的内部本质上就是 set\_item、get\_item 方法。

基础知识:

索引的语法:

public string this[string s], 通过 get、set 块来定义取值、赋值的逻辑

索引可以有多个参数、参数类型任意

索引可以重载。

如果只有 get 没有 set 就是只读的索引。

索引其实就是 set\_Item、get\_Item 两个方法。

**7、属性和 public 字段的区别是什么？** 调用 set 方法为一个属性设值，然后用 get 方法读取出来的值一定是 set 进去的值吗？

属性可以对设值、取值的过程进行非法值控制，比如年龄禁止设值负数，而字段则不能进行这样的设置。虽然一般情况下 get 读取的值就是 set 设置的值，但是可以让 get 读取的值不是 set 设置的值的，**极端的例子**。Public Age{get{return 100;}set{}}。加分的补充回答：用 reflector 反编译可以看出，**属性内部本质上就是 set\_\*\*\*、get\_\*\*\*方法**，详细参考传智播客.net 培训视频中串讲.net 基础的部分。

```
class Person
{
    public int Age
    {
        get
        {
            return 3;
        }
        set
        {
        }
    }
}

Person p1 = new Person();
p1.Age = 30;
p1.Age++;
Console.Write(p1.Age); //输出 3
```

## 8、三层架构

通常意义上的三层架构就是将整个业务应用划分为：**表现层（UI）、业务逻辑层（BLL）、数据访问层（DAL）**。

区分层次的目的即为了“高内聚，低耦合”的思想。

**表现层（UI）**：通俗讲就是展现给用户的界面，即用户在使用一个系统的时候的所见所得。

**业务逻辑层（BLL）**：针对具体问题的操作，也可以说是对数据层的操作，对数据业务逻辑处理。

**数据访问层（DAL）**：该层所做事务直接操作数据库，针对数据的增添、删除、修改、更新、查找等每层之间是一种垂直的关系。

三层结构是 N 层结构的一种，一般来说，层次之间是向下依赖的，下层代码未确定其接口（契约）前，上层代码是无法开发的，下层代码接口（契约）的变化将使上层的代码一起变化。

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。

## 10、关于拆箱装箱：

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

### 1) 什么是装箱(boxing)和拆箱(unboxing)? (\*)

Object 是引用类型，但是它的子类 Int32 竟然不能去 Object 能去的“要求必须是引用类型”

的地方，违反了继承的原则，所以需要把 Int32 装在 Object 中才能传递。

装箱：从值类型接口转换到引用类型。

拆箱：从引用类型转换到值类型。

```
object obj = null; //引用类型
obj = 1; //装箱, boxing. 把值类型包装为引用类型。
int i1 = (int)obj; //拆箱. unboxing
```

2) 下面三句代码有没有错，以 inboxing 或者 unboxing 为例，解释一下内存是怎么变化的

```
int i=10;
```

```
object obj = i;
```

```
int j = obj;
```

分析：在 inboxing（装箱）时是不需要显式的类型转换的，不过 unboxing（拆箱）需要显式的类型转换，所以第三行代码应该改为：

```
3 int j = (int)obj;
```

要掌握装箱与拆箱，就必须了解 CTS 及它的特点：

NET 重要技术和基础之一的 CTS(Common Type System)。CTS 是为了实现在应用程序声明和使用这些类型时必须遵循的规则而存在的通用类型系统。.Net 将整个系统的类型分成两大类：值类型和引用类型。

CTS 中的所有东西都是对象；所有的对象都源自一个基类——System.Object 类型。值类型的一个最大的特点是它们不能为 null，值类型的变量总有一个值。为了解决值类型不可以为 null，引用类型可以为 null 的问题，微软在 .Net 中引入了装箱和拆箱：装箱就是将值类型用引用类型包装起来转换为引用类型；而从引用类型中拿到被包装的值类型数据进行拆箱。

```
object.ReferenceEquals();//用来判断两个对象是否是同一个对象
```

```
Console.WriteLine(object.ReferenceEquals(3,3)); //因为两个3被装到了两个箱子中，所以是false
```

Equals == 的关系

### 12、CTS、CLS、CLR 分别作何解释 (\*) 把英文全称背过来。

C#和.Net 的关系。

C#只是抽象的语言，可以把 C#编译生成 Java 平台的二进制代码，也可以把 Java 代码编译生成 .Net 平台的二进制代码。所以 C#只是提供了 if、while、+、\*、/、定义类、int、string 等基础的语法，而 Convert.ToInt32、FileStream、SqlConnection、String.Split 等都属于 .Net 的东西。深蓝色是 C#的，浅蓝色是 .Net 的。

C# new → IL: newobj

C# string → .Net 中的 String

类型的差别：.net 中的 Int32 在 C#中是 int，在 VB.Net 中是 Integer。

语法的差别：IL 中创建一个对象的方法是 L\_0001: newobj instance void 索引.C1::ctor()

C#中是 new C1(); VB.net 中是 Dim c1 As New C1

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

**CTS: Common Type System** 通用类型系统。Int32、Int16→int、String→string、Boolean→bool。每种语言都定义了自己的类型，.Net 通过 CTS 提供了公共的类型，然后翻译生成对应的.Net 类型。

**CLS: Common Language Specification** 通用语言规范。不同语言语法的不同。每种语言都有自己的语法，.Net 通过 CLS 提供了公共的语法，然后不同语言翻译生成对应的.Net 语法。

**CLR: Common Language Runtime** 公共语言运行时，就是 GC、JIT 等这些。有不同的 CLR，比如服务器 CLR、Linux CLR (Mono)、Silverlight CLR(CoreCLR)。相当于一个发动机，负责执行 IL。

### 13、在 dotnet 中类(class)与结构(struct)的异同？

Class 可以被实例化,属于引用类型,是分配在内存的堆上的。类是引用传递的。

Struct 属于值类型,是分配在内存的栈上的。结构体是复制传递的。加分的回答: Int32、Boolean 等都属于结构体。

### 14、堆和栈的区别？

栈是编译期间就分配好的内存空间，因此你的代码中必须就栈的大小有明确的定义；局部值类型变量、值类型参数等都在栈内存中。

堆是程序运行期间动态分配的内存空间，你可以根据程序的运行情况确定要分配的堆内存的大小。

### 15、能用 foreach 遍历访问的对象的要求

需要实现 IEnumerable 接口或声明 GetEnumerator 方法的类型。

### 16、GC 是什么？为什么要有 GC？

C/C++中由程序员进行对象的回收像学校食堂中由学生收盘子，.Net 中由 GC 进行垃圾回收像餐馆中店员去回收。

GC 是垃圾收集器 (Garbage Collection)。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。GC 只能处理托管内存资源的释放，对于非托管资源则不能使用 GC 进行回收，必须由程序员手工回收，一个例子就是 FileStream 或者 SqlConnection 需要程序员调用 Dispose 进行资源的回收。

要请求垃圾收集，可以调用下面的方法：GC.Collect()一般不需要手动调用 GC.Collect()。当一个对象没有任何变量指向（不再能使用）的时候就可以被回收了。

基础知识：当没有任何变量指向一个对象的时候对象就可以被回收掉了，但不一定会立即被回收。

```
object obj = new object();//只有new才会有新对象
Console.WriteLine(obj);
```

```
object obj2 = obj;
obj = null;
obj2 = null;
Console.WriteLine();
```



## 18、值类型和引用类型的区别？

1. 将一个值类型变量赋给另一个值类型变量时，将复制包含的值。引用类型变量的赋值只复制对象的引用，而不复制对象本身。
2. 值类型不可能派生出新的类型：所有的值类型均隐式派生自 `System.ValueType`。但与引用类型相同的是，结构也可以实现接口。
3. 值类型不可能包含 `null` 值：然而，可空类型功能允许将 `null` 赋给值类型。
4. 每种值类型均有一个隐式的默认构造函数来初始化该类型的默认值。

## 19、C#中的接口和类有什么异同。

不同点：

不能直接实例化接口。

接口不包含方法的实现。

接口可以多继承，类只能单继承。

类定义可在不同的源文件之间进行拆分。

相同点：

接口、类和结构都可以从多个接口继承。

接口类似于抽象基类：继承接口的任何非抽象类型都必须实现接口的所有成员。

接口和类都可以包含事件、索引器、方法和属性。

基础知识：接口只能定义方法（只能定义行为，不能定义实现也就是字段），因为事件、索引器、属性本质上都是方法，所以接口中也可以定义事件、索引器、属性。

## 20、abstract class 和 interface 有什么区别？

相同点：

都不能被直接实例化，都可以通过继承实现其抽象方法。

不同点：

接口支持多继承；抽象类不能实现多继承。

接口只能定义行为；抽象类既可以定义行为，还可能提供实现。

接口只包含方法(Method)、属性(Property)、索引器(Index)、事件(Event)的签名，但不能定义字段和包含实现的方法；

抽象类可以定义字段、属性、包含有实现的方法。

接口可以作用于值类型(Struct)和引用类型(Class)；抽象类只能作用于引用类型。例如，Struct 就可以继承接口，而不能继承类。

加分的补充回答：讲设计模式的时候 SettingsProvider 的例子。

## 21、是否可以继承 String 类？

String 类是 sealed 类故不可以继承。

## 22、

```
static void Main(string[] args)
{
    Console.WriteLine(GetIt());
}
```

```
        Console.ReadKey();
    }

    static int GetIt()
    {
        int i = 8;
        try
        {
            i++;
            Console.WriteLine("a");
            return i; //把返回值设定为i，然后“尽快”返回（没啥事就回去吧）
        }
        finally
        {
            Console.WriteLine("b");
            i++;
        }
    }
}
```

上面程序的执行结果是 ab9

**try {}**里有一个 **return** 语句，那么紧跟在这个 **try** 后的 **finally {}**里的代码会不会被执行，什么时候被执行？

会执行，在 **return** 后执行。

```
static void Main(string[] args)
{
    //Console.WriteLine(GetIt());
    Console.WriteLine(GetPerson().Age);
    Console.ReadKey();
}

static int GetIt()
{
    int i = 8;
    try
    {
        i++;
        Console.WriteLine("a");
        return i; //把返回值设定为i，然后“尽快”返回（没啥事就回去吧。搞完就走）
    }
    finally
    {
        Console.WriteLine("b");
        i++;
    }
}
```

```
    }  
}  
  
static Person GetPerson()  
{  
    Person p = new Person();  
    p.Age = 8;  
    try  
    {  
        p.Age++;  
        Console.WriteLine("a");  
        return p; //把返回值设定为i, 然后“尽快”返回（没啥事就回去吧。搞完就走）  
    }  
    finally  
    {  
        Console.WriteLine("b");  
        p.Age++;  
    }  
}
```

加分的补充回答（也助记）：读取数据库中数据的条数的程序

```
public int QueryCount()  
{  
    .....  
    try  
    {  
        return cmd.ExecuteScalar();  
    }  
    finally  
    {  
        cmd.Dispose();  
    }  
}
```

先执行 `cmd.ExecuteScalar()`，把返回值暂时存起来，然后再去执行 `finally`（钱放在这，我去劫个色），然后把返回值返回。`return` 都是最后执行，但是 `return` 后的表达式的计算则是在 `finally` 之前。

如果 C# 设计的是先执行 `cmd.Dispose()` 再执行 `return` 就会出现 `return` 执行失败了，因为 `cmd` 已经 `Dispose` 了。

**28、int、DateTime、string 是否可以为 null?**

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

**null** 表示“不知道”，而不是“没有”。没有年龄和不知道年龄是不一样的。

数据库中 **null** 不能用 **0** 表示。**0** 岁和不知道多少岁不一样。

**Sex is zero.**《色即是空》

```
//int i1 = null;  
//int? i2 = null; //值类型后加?就成了可空数据类型  
////int i3 = i2; //所以把可空的赋值给一定不能为空的会报错。  
//int i4 = (int)i2; //可以显式转换，由程序员来保证“一定不为空”  
//int? i5 = i4; //一定会成功!
```

```
//using()→try{}catch{}finally{}
```

//int?是微软的一个语法糖。是一种和int没有直接关系的Nullable类型

```
Nullable<int> d1 = new Nullable<int>(); //int? d1=null;  
Nullable<int> d2 = new Nullable<int>(3); //int? d2=3;  
Console.WriteLine(d1==null);
```

int、DateTime 不能，因为其为 Struct 类型，而结构属于值类型，值类型不能为 null，只有引用类型才能被赋值 null。string 可以为 null。

C#中 int 等值类型不可以为 null、而数据库中的 int 可以为 null，这就是纠结的地方。int?就变成了可空的 int 类型。bool?、DateTime?

```
int i1 = 3;  
int? i2 = i1;  
//int i3 = i2; //不能把可能为空的赋值给一定不能为空的变量  
int i3 = (int)i2; //显式转换
```

可空数据类型经典应用场景：三层中的 Model 属性的设计。

int?翻译成.Net 的 Nullable<int>，CTS。

### 29、using 关键字有什么用？什么是 IDisposable？

using 可以声明 namespace 的引入，还可以实现非托管资源的释放，实现了 IDisposable 的类在 using 中创建，using 结束后会自动调用该对象的 Dispose 方法，释放资源。加分的补充回答：using 其实等价于 try……finally，用起来更方便。

### 30、XML 与 HTML 的主要区别

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。

HTML: <br><br>

XML: </img><br/><br/>

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。

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

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。

XML是用来存储和传输数据的

HTML 是用来显示数据的

如果使用了完全符合 XML 语法要求的 HTML,那么就叫做符合 XHTML 标准。符合 XHTML 标准的页面有利于 SEO。

**31、string str = null 与 string str = ""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。 string str = null 没有 string 对象，string str = ""有一个字符串对象。

```
string s3 = string.Empty;//反编译发现，string.Empty就是在类构造函数中 Empty = "";
```

**32.写出一条 Sql 语句：取出表 A 中第 31 到第 40 记录 (SQLServer,以自动增长的 ID 作为主键,注意：ID 可能不是连续的。**

答：解 1: select top 10 \* from A where id not in (select top 30 id from A)

演变步骤：

```
1) select top 30 id from T_FilterWords--取前条
```

```
2) select * from T_FilterWords
```

```
where id not in (select top 30 id from T_FilterWords)--取id不等于前三十  
条的  
--也就是把前条排除在外
```

```
3) select top 10 * from T_FilterWords
```

```
where id not in (select top 30 id from T_FilterWords)  
--取把前条排除在外的前条，也就是-40 条
```

解 2: select top 10 \* from A where id > (select max(id) from (select top 30 id from A )as A)

解答 3：用 ROW\_NUMBER 实现

**33.面向对象的语言具有\_\_\_\_\_性、\_\_\_\_\_性、\_\_\_\_\_性**

答：封装、继承、多态。

不要背，脑子中要有联想。

**34.在.Net 中所有可序列化的类都被标记为\_\_\_\_\_?**

答：[serializable]

**35、什么是 code-Behind 技术。**

就是代码隐藏，在 ASP.NET 中通过 ASPX 页面指向 CS 文件的方法实现显示逻辑和处理逻辑的分离，这样有助于 web 应用程序的创建。比如分工，美工和编程的可以个干各的，不用再像以前 asp 那样都代码和 html 代码混在一起，难以维护。**code-Behind 是基于部分类技术实现的，在我的项目的三层代码生成器中用到了部分类。**



36、接口是一种引用类型，在接口中可以声明（ a ），但不可以声明公有的域或私有的成员变量。

- a) 方法、属性、索引器和事件；
- b) 索引器和字段；
- c) 事件和字段；

解读：接口中不能声明字段只能声明方法，属性、索引器和事件 最终都编译生成方法。因为字段属于实现层面的东西，只有存取值的时候才会用到字段，所以中接口中不能定义字段。

37. 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）。

- a) insert、update、delete 等操作的 Sql 语句主要用 ExecuteNonQuery() 方法来执行；
- b) ExecuteNonQuery()方法返回执行 Sql 语句所影响的行数。
- c) Select 操作的 Sql 语句只能由 ExecuteReader()方法来执行；
- d) ExecuteReader()方法返回一个 DataReader 对象；

拿 SQLHelper 实现一下。

38. 下列关于 C#中索引器理解正确的是(c )

- a) 索引器的参数必须是两个或两个以上
- b) 索引器的参数类型必须是整数型
- c) 索引器没有名字
- d) 以上皆非

39. 下面关于 XML 的描述错误的是（d）。

- a) XML 提供一种描述结构化数据的方法；
- b) XML 是一种简单、与平台无关并被广泛采用的标准；
- c) XML 文档可承载各种信息；
- d) XML 只是为了生成结构化文档；

40、以下的 C#代码，试图用来定义一个接口：

```
public interface IFile
{
    int A;
    int delFile()
    {
        A = 3;
    }
    void disFile();
}
```

关于以上的代码，以下描述错误的是(d )。

- a) 以上的代码中存在的错误包括：不能在接口中定义变量，所以 int A 代码行将出现错误；
- b) 以上的代码中存在的错误包括：接口方法 delFile 是不允许实现的，所以不能编写具体

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

---

的实现函数;

- c) 代码 `void disFile()`;声明无错误, 接口可以没有返回值;
- d) 代码 `void disFile()`;应该编写为 `void disFile(){};`

41、在 ASP.NET 中有 Button 控件 myButton, 要是单击控件时, 导航到其他页面 <http://www.abc.com>, 正确的代码为( c)。

- a)  
`private void myButton_Click(object sender, System.EventArgs e){Redirect("http://www.abc.com");}`
- b)  
`private void myButton_Click(object sender, System.EventArgs e){Request.Redirect("http://www.abc.com");}`
- c)  
`private void myButton_Click(object sender, System.EventArgs e){Response.Redirect("http://www.abc.com");}`
- d)  
`private void myButton_Click(object sender, System.EventArgs e){Request.Redirect("http://www.abc.com");return true;}`

42. 声明一个委托 `public delegate int myCallBack(int x);` 则用该委托产生的回调方法的原型应该是 (b )。

- a) `void myCallBack(int x) ;`
- b) `int receive(int num) ;`
- c) `string receive(int x) ;`
- d) 不确定的;

43.StringBuilder 和 String 的区别?

答: **String** 在进行运算时 (如赋值、拼接等) 会产生一个新的实例, 而 **StringBuilder** 则不会。所以在大量字符串拼接或频繁对某一字符串进行操作时最好使用 **StringBuilder**, 不要使用 **String**

如果要操作一个不断增长的字符串, 尽量不用 **String** 类, 改用 **StringBuilder** 类。两个类的工作原理不同: **String** 类是一种传统的修改字符串的方式, 它确实可以完成把一个字符串添加到另一个字符串上的工作没错, 但是在 .NET 框架下, 这个操作实在是划不来。因为系统先是将两个字符串写入内存, 接着删除原来的 **String** 对象, 然后创建一个 **String** 对象, 并读取内存中的数据赋给该对象。这一来二去的, 耗了不少时间。而使用 **System.Text** 命名空间下面的 **StringBuilder** 类就不是这样了, 它提供的 **Append** 方法, 能够在已有对象的原地进行字符串的修改, 简单而且直接。当然, 一般情况下觉察不到这二者效率的差异, 但如果你要对某个字符串进行大量的添加操作, 那么 **StringBuilder** 类所耗费的时间和 **String** 类简直不是一个数量级的。

44、.请叙述属性与索引器的区别。 (\*)

属性 索引器

通过名称标识。 通过签名标识。

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

通过简单名称或成员访问来访问。通过元素访问来访问。

可以为静态成员或实例成员。必须为实例成员。

属性的 `get` 访问器没有参数。索引器的 `get` 访问器具有与索引器相同的形参表。

属性的 `set` 访问器包含隐式 `value` 参数。除了 `value` 参数外，索引器的 `set` 访问器还具有与索引器相同的形参表。

### 46. 请解释 ASP.NET 中的 web 页面与其隐藏类之间的关系？

一个 ASP.NET 页面一般都对应一个隐藏类，一般都在 ASP.NET 页面的声明中指定了隐藏类例  
如一个页面 `Tst1.aspx` 的页面声明如下

```
<%@ Page language="c#" Codebehind="Tst1.aspx.cs" AutoEventWireup="false"
Inherits="T1.Tst1" %>
```

`Codebehind="Tst1.aspx.cs"` 表明经编译此页面时使用哪一个代码文件

`Inherits="T1.Tst1"` 表用运行时使用哪一个隐藏类

`aspx` 页面会编译生成一个类，这个类从隐藏类继承。

### 49. 您在什么情况下会用到虚方法？它与接口有什么不同？

答案：子类重新定义父类的某一个方法时，必须把父类的方法定义为 `virtual`

在定义接口中不能有方法体，虚方法可以。

实现时，子类可以不重新定义虚方法，但如果一个类继承接口，那必须实现这个接口。

### 50. 不定项选择：

(1) 以下叙述正确的是： B C

A. 接口中可以有虚方法。 B. 一个类可以实现多个接口。 C. 接口不能被实例化。 D. 接口中可以包含已实现的方法。

(2) 从数据库读取记录，你可能用到的方法有： B C D

A. `ExecuteNonQuery` B. `ExecuteScalar` C. `Fill` D. `ExecuteReader`

(3) 对于一个实现了 `IDisposable` 接口的类，以下哪些项可以执行与释放或重置非托管资源相关的应用程序定义的任务？(多选) ( ABCD )

A. `Close` B. `Dispose` C. `Finalize`

D. `using` E. `Quit`

(4) 以下关于 `ref` 和 `out` 的描述哪些项是正确的？(多选) ( ACD )

A. 使用 `ref` 参数，传递到 `ref` 参数的参数必须最先初始化。

B. 使用 `out` 参数，传递到 `out` 参数的参数必须最先初始化。

C. 使用 `ref` 参数，必须将参数作为 `ref` 参数显式传递到方法。

D. 使用 `out` 参数，必须将参数作为 `out` 参数显式传递到方法。

### 51. 单项选择：

(1) 下列选项中，(C) 是引用类型。

a) `enum` 类型 b) `struct` 类型 c) `string` 类型 d) `int` 类型

(2) 关于 ASP.NET 中的代码隐藏文件的描述正确的是 (C)

a) Web 窗体页的程序的逻辑由代码组成，这些代码的创建用于与窗体交互。编程逻辑唯一与

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

用户界面不同的文件中。该文件称作为“代码隐藏”文件，如果用 C# 创建，该文件将具有“.ascx.cs”扩展名。

b) 项目中所有 Web 窗体页的代码隐藏文件都被编译成 EXE 文件。

c) 项目中所有的 Web 窗体页的代码隐藏文件都被编译成项目动态链接库 (.dll) 文件。

d) 以上都不正确。

(3) . 下列描述错误的是 (D)

a) 类不可以多重继承而接口可以；

b) 抽象类自身可以定义成员而接口不可以；

c) 抽象类和接口都不能被实例化；

d) 一个类可以有多个基类和多个基接口；

### 52、DataReader 和 DataSet 的异同？

DataReader 使用时始终占用 SqlConnection, 在线操作数据库

每次只在内存中加载一条数据, 所以占用的内存是很小的

是只进的、只读的

DataSet 则是将数据一次性加载在内存中. 抛弃数据库连接.. 读取完毕即放弃数据库连接(非连接模式)

DataSet 将数据全部加载在内存中. 所以比较消耗内存... 但是确比 DataReader 要灵活.. 可以动态的添加行, 列, 数据. 对数据库进行 回传更新操作(动态操作读入到内存的数据)

### 53、public static const int A = 1; 这段代码有错误么？

错误: const 不能被修饰为 static ; 因为定义为常量 (const ) 后就是静态的 (static )。

### 54、传入某个属性的 set 方法的隐含参数的名称是什么？

value, 它的类型和属性所声明的类型相同。

### 55、C# 支持多重继承么？

类之间不支持，接口之间支持。类对接口叫做实现，不叫继承。类是爹、接口是能力，能有多能力，但不能有多爹。

### 56、C# 中所有对象共同的基类是什么？

System.Object

### 57、通过超链接怎样传递中文参数？

答: 用 URL 编码, 通过 QueryString 传递, 用 urlencode 编码 用 urldecode 解码

### 58、string、String; int、Int32; Boolean、bool 的区别

String、Int32、Boolean 等都属于 .Net 中定义的类, 而 string、int、bool 相当于 C# 中对这些类定义的别名。CTS。

### 59、Server.Transfer 和 Response.Redirect 的区别是什么？(常考)

答: **Server.Transfer** 仅是服务器中控制权的转向, 在客户端浏览器地址栏中不会显示出转向后的地址; **Response.Redirect** 则是完全的跳转, 浏览器将会得到跳转的地址, 并重新发送请

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

求链接。这样，从浏览器的地址栏中可以看到跳转后的链接地址。

Server.Transfer 是服务器请求资源，服务器直接访问目标地址的 URL，把那个 URL 的响应内容读取过来，然后把这些内容再发给浏览器，浏览器根本不知道服务器发送的内容是从哪儿来的，所以它的地址栏中还是原来的地址。这个过程中浏览器和 Web 服务器之间经过了一次交互。

Response.Redirect 就是服务端根据逻辑，发送一个状态码，告诉浏览器重新去请求那个地址，一般来说浏览器会用刚才请求的所有参数重新请求。这个过程中浏览器和 Web 服务器之间经过了两次交互。

Server.Transfer 不可以转向外部网站，而 Response.Redirect 可以。

### 60、不是说字符串是不可变的吗？string s="abc";s="123"不就是变了吗？（传智播客.Net 培训班原创模拟题）

String 是不可变的在这段代码中，s 原先指向一个 String 对象，内容是 "abc"，然后我们将 s 指向 "123"，那么 s 所指向的那个对象是否发生了改变呢？答案是没有。这时，s 不指向原来那个对象了，而指向了另一个 String 对象，内容为 "123"，原来那个对象还存在于内存之中，只是 s 这个引用变量不再指向它了。

### 61、是否可以从一个 static 方法内部发出对非 static 方法的调用？

不可以。因为非 static 方法是要与对象关联在一起的，必须创建一个对象后，才可以在该对象上进行方法调用，而 static 方法调用时不需要创建对象，可以直接调用。也就是说，当一个 static 方法被调用时，可能还没有创建任何实例对象，如果从一个 static 方法中发出对非 static 方法的调用，那个非 static 方法是关联到哪个对象上的呢？这个逻辑无法成立，所以，一个 static 方法内部不能发出对非 static 方法的调用。

### 63、说出一些常用的类、接口，请各举 5 个

要让人家感觉你对 .Net 开发很熟，所以，不能仅仅只列谁都能想到的那些东西，要多列你在做项目中涉及的那些东西。就写你最近写的那些程序中涉及的那些类。

常用的类：StreamReader、WebClient、Dictionary<K,V>、StringBuilder、SqlConnection、FileStream、File、Regex、List<T>

常用的接口：IDisposable、IEnumerable、IDbConnection、IComparable、ICollection、IList、IDictionary

要出乎意料！不要仅仅完成任务！笔试不是高考！处处要显出牛！

### 65、编写一个单例（Singleton）类。

参考复习 ppt。

把构造函数设置为 private，设置一个 public、static 的对象实例

```
public FileManager
{
    private FileManager(){}
    public readonly static FileManager Instance = new FileManager();
}
```

扩展：搜“C# Singleton”，有线程安全的更牛 B 的实现



## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

---

### 66、什么是 sql 注入？如何避免 sql 注入？

用户根据系统的程序构造非法的参数从而导致程序执行不是程序员期望的恶意 SQL 语句。  
使用参数化的 SQL 就可以避免 SQL 注入。

详细参考复习 ppt。举例子，摆事实！

1' or 1=1

### 67、数据库三范式是什么？

用自己的话解释，而不是背概念。

传智播客风格的非严谨却一下子明了的解答方式：

第一范式：字段不能有冗余信息，所有字段都是必不可少的。

第二范式：满足第一范式并且表必须有主键。

第三范式：满足第二范式并且表引用其他的表必须通过主键引用。

记忆顺序：自己内部不重复→别人引用自己→自己引用别人。

1) 不一定满足第一范式，比如拼音检索中增加一个“拼音”字段。

Id	Name	Pinyin
1	杨中科	yangzhongke
2	杨开慧	yangkaihui
3	赵美美	zhaomeimei

whwere pinyin like 'yang%'

### 72、post、get 的区别

get 的参数会显示在浏览器地址栏中，而 post 的参数不会显示在浏览器地址栏中；

使用 post 提交的页面在点击【刷新】按钮的时候浏览器一般会提示“是否重新提交”，而 get 则不会；

用 get 的页面可以被搜索引擎抓取，而用 post 的则不可以；

用 post 可以提交的数据量非常大，而用 get 可以提交的数据量则非常小(2k)，受限于网页地址的长度。

用 post 可以进行文件的提交，而用 get 则不可以。

参考阅读：<http://www.cnblogs.com/skynet/archive/2010/05/18/1738301.html>

### 73、下面的程序执行结果是什么？（传智播客.Net 培训班原创模拟题）

```
class Person
{
    public int Age { get; set; }
}

int i1 = 20;
int i2 = i1;
i1++;
Console.WriteLine(i2);
```

```
Person p1 = new Person();  
p1.Age = 20;  
Person p2 = p1;  
p1.Age++;  
Console.WriteLine(p2.Age);
```

答案：

20

21

解释：引用类型和值类型的区别。

76、下面程序的执行结果是什么？（传智播客.Net 培训班原创模拟题）

```
int i = 10;  
Console.WriteLine(i++);  
Console.WriteLine(++i);  
Console.WriteLine(i=20);  
Console.WriteLine(i==20);
```

答案：

10

12

20

True

解答：前两个参考第 80 题，`i++`表达式的值为 `i` 自增之前的值，所以打印 10，执行完成后 `i` 变成 11，执行 `++i` 的时候表达式的值为自增以后的值 12。C# 中赋值表达式的值为变量被赋值的值，因此最后一个就输出 `i` 赋值后的值 20

```
while((line=reader.ReadLine())!=null)  
{  
}
```

下面程序的执行结果是什么？

```
bool b=false;  
if(b=true)  
{  
    Console.WriteLine("yes");  
}  
else  
{  
    Console.WriteLine("no");  
}
```

78、下列哪个名词和 WebService 不直接相关？（B）（传智播客.Net 培训班原创模拟题）

A、UDDI

B、GUID

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 net.ITHeima.com

---

C、WSDL

D、SOAP

UDDI 用来自动发现 WebService 的协议 (\*)

SOAP 和 HTTP 的关系: SOAP 基于 Http 协议的, 和普通网页不同的是网页返回 HTML, SOAP 则是符合 SOAP 协议的 XML 数据。

WSDL(WebService Defination Language)是对 WebService 上的方法名、参数进行描述的协议

SOAP(Simple Object Access Protocol)是对参数、返回值以什么样的格式进行传递进行描述的协议。

81、.Net、ASP.Net、C#、VisualStudio 之间的关系是什么? (传智播客.Net 培训班原创模拟题)

答: .Net 一般指的是 .Net Framework, 提供了基础的 .Net 类, 这些类可以被任何一种 .Net 编程语言调用, .Net Framework 还提供了 CLR、JIT、GC 等基础功能。ASP.Net 是 .Net 中用来进行 Web 开发的一种技术, ASP.Net 的页面部分写在 aspx 文件中, 逻辑代码通常通过 Code-behind 的方式用 C#、VB.Net 等支持 .Net 的语言编写。

C#是使用最广泛的支持 .Net 的编程语言。除了 C#还有 VB.Net、IronPython 等。VisualStudio 是微软提供的用来进行 .Net 开发的集成开发环境 (IDE), 使用 VisualStudio 可以简化很多工作, 不用程序员直接调用 csc.exe 等命令行进行程序的编译, 而且 VisualStudio 提供了代码自动完成、代码高亮等功能方便开发。除了 VisualStudio, 还有 SharpDevelop、MonoDevelop 等免费、开源的 IDE, VisualStudio Express 版这个免费版本。

82、AJAX 解决什么问题? 如何使用 AJAX? AJAX 有什么问题需要注意? 项目中哪里用到了 AJAX? (传智播客.Net 培训班原创模拟题)

答: AJAX 解决的问题就是“无刷新更新页面”, 用传统的 HTML 表单方式进行页面的更新时, 每次都要将请求提交到服务器, 服务器返回后再重绘界面, 这样界面就会经历: 提交→变白→重新显示这样一个过程, 用户体验非常差, 使用 AJAX 则不会导致页面重新提交、刷新。

AJAX 最本质的实现是在 Javascript 中使用 XMLHttpRequest 进行 Http 的请求, 开发中通常使用 UpdatePanel、JQuery 等方式简化 AJAX 的开发, UpdatePanel 的方式实现 AJAX 最简单, 但是数据通讯量比较大, 因为要来回传整个 ViewState, 而且不灵活, 对于复杂的需求则可以使用 JQuery 提供的 ajax 功能。

UpdatePanel 的内部原理。

AJAX 最重要的问题是无法跨域请求([www.rupeng.com](http://www.rupeng.com) → [so.rupeng.com](http://so.rupeng.com)), 也就是无法在页面中向和当前域名不同的页面发送请求, 可以使用在当前页面所在的域的服务端做代理页面的方式解决。

在如鹏网项目中发帖的时候显示相关帖的功能、站内搜索项目中显示搜索 Suggestion、数据采集项目中都用到了 AJAX。

常考: 不用任何框架编写一个 AJAX 程序。XHR: XMLHttpRequest。

如果面试的时候谈 AJAX 谈到 UpdatePanel 的时候，就是 NB 的时候！！！先侃 UpdatePanel 的原理！引出为什么 Dom 操作的动态效果在用 UpdatePanel 提交刷新以后没有了，以及 CKEditor 被套在 UpdatePanel 中提交以后也变成了 textarea，为什么把 Fileupload 放到 Updatepanel 中无法实现无刷新上传。说成是公司内部的一个菜鸟用 UpdatePanel 遇到这样问题，由于我懂 XHR、UpdatePanel 的原理，所以轻松解决！UpdatePanel 生成的上万行 JS 脚本，不适合于互联网项目。“WebForm 怎么可能把开发人员编程傻子呢！不明白原理苦命呀！”

多准备几把斧子！上来找一切机会甩斧子！

84、Application、Cookie 和 Session 两种会话有什么不同？

答：Application 是用来存取整个网站全局的信息，而 Session 是用来存取与具体某个访问者关联的信息。Cookie 是保存在客户端的，机密信息不能保存在 Cookie 中，只能放小数据；Session 是保存在服务器端的，比较安全，可以放大数据。

谈到 Session 的时候就侃 Session 和 Cookie 的关系：Cookie 中的 SessionId。和别人对比说自己懂这个原理而给工作带来的方便之处。

85、开放式问题：你经常访问的技术类的网站是什么？

博客园（[www.cnblogs.com](http://www.cnblogs.com)）、csdn、codeplex、codeproject、msdn 文档、msdn 论坛（遇到问题先到网上搜解决方案，还不行就问同事，同事也解决不了就去 MSDN 论坛提问，一定能得到解决）。cnbeta。

86、你对.net 的 GC 的理解

GC 是 .Net 的垃圾收集器，可以进行内存资源的回收，程序员无需关心资源的回收，当一个对象没有任何引用的时候就可以被回收了。一个对象可以被回收并不意味着一定会被立即回收，GC 会选择时机进行回收。可以调用 GC.Collect() 让 GC 立即回收。GC 不能回收非托管资源，对于非托管资源一般都实现了 IDisposable 接口，然后使用 using 关键字进行资源的回收。

87、请写一个 SQL 语句：从 user 表中取出 name 列中的起始字符是“北京”的全部记录

```
select * from [user] wherer name like '北京%'
```

88、请你简单的说明数据库建立索引的优缺点

使用索引可以加快数据的查询速度，不过由于数据插入过程中会建索引，所以会降低数据的插入、更新速度，索引还会占磁盘空间。

101、下面程序的执行结果是什么？

```
public struct Point
```

```
{
    public int x;
    public int y;
    public Point(int x, int y)
    {
        this.x = x;
        this.y = y;
    }
}

Point p1 = new Point(100, 100);
Point p2 = p1;
p1.x = 200;
Console.WriteLine("{0},{1}", p1.x, p2.x);
```

答案：

200,100

解答：结构体是复制传递的。

扩展：为啥`this.Size.Width = 100`;编译不通过。

106、不用 JQuery 等框架实现 AJAX 无刷新登录。（常考）

提示：用纯 XMLHttpRequest 实现。

107、Http 状态码各是什么意思。

301（永久重定向）、302：重定向

404：页面不存在

500：服务器内部错误

108、对 html 文本框进行赋值的 JQuery 语句,对 html 文本框进行取值的 JQuery 语句,对 TextBox 服务端控件进行取值的语句 JQuery。

110、Session 有什么重大 BUG，微软提出了什么方法加以解决？（常考）

iis 中由于有进程回收机制，系统繁忙的话 Session 会丢失，IIS 重启也会造成 Session 丢失。这样用户就要重新登录或者重新添加购物车、验证码等放到 Session 中的信息。可以用 State 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。但是（一定要写上，这样才够拽）我认为这不是 Bug，只能说是 In-Proc 方式存储 Session 的缺陷，缺陷是和 Bug 不一样的，In-Proc 方式存储 Session 会由服务器来决定什么时候释放 Session，这是 By Design，In-Proc 方式不满足要求的话完全可以用 StateServer 和数据库的方式。

StateServer 还可以解决集群 Session 共享的问题。



不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 net.ITHeima.com

---

配置 StateServer 的方法：

<http://support.microsoft.com/kb/317604>

<http://blog.csdn.net/zj1103/archive/2008/09/14/2926773.aspx>

111、asp.net 中<%%>、<%= %>、<%# %>的区别是什么

<%%>是执行<%%>中的 C#代码，<%= %>是将=后表达式的值输出到 Response 中，<%# %>是数据绑定，一般用在 ListView、GridView、Repeater 等控件的绑定中。数据绑定分为：Eval：单向绑定和 Bind：双向绑定。

113、说说在软件设计中你遇到的以空间换时间的例子？

参考回答：谈 Cache(缓存)、索引这些例子。用额外的磁盘、内存空间的消耗来提高执行速度。Lucene.Net 也是一个例子。

114、asp.net 中的错误机制。（常考）

参考：

1) 定制错误页来将显示一个友好的报错页面。

页面中未捕获一样会触发 Page\_Error（不常用），应用程序中的未捕获异常会触发 Application\_Error。通过

HttpContext.Current.Server.GetLastError()、HttpContext.Current.Error;拿到未捕捉异常，记录到Log4Net日志中。

115、不用中间变量交换两个变量

```
int i = 500;
int j = int.MaxValue - 10;
//int i = 10;
//int j = 20;

Console.WriteLine("i={0},j={1}", i, j);

i = i + j; //i=30
j = i - j; //j=10;
i = i - j; //i=20;

Console.WriteLine("i={0},j={1}", i, j);
```

另外一个解决方案：位运算。

```
int a=5;
int b=6;
a=a^b;
b=b^a; //b^a 相当于 b^a^b 也就是 b^a^b 的值就是 a 了，下边相同
a=a^b;
```

下面的 SQL 题是常考题中的常考题，必须重视！！！！！！！！

## 116、横表、纵表转换

### 1) 纵表结构 TableA

Name	Course	Grade
张三	语文	75
张三	数学	80
张三	英语	90
李四	语文	95
李四	数学	55

### 横表结构 TableB

Name	语文	数学	英语
张三	75	80	90
李四	95	55	0

先理解：

`select Name,`

`(case Course when '语文' then Grade else 0 end) as 语文,`

`(case Course when '数学' then Grade else 0 end) as 数学,`

`(case Course when '英语' then Grade else 0 end) as 英语`

`from TableA`

然后理解标准答案：

`select Name,`

`sum(case Course when '语文' then Grade else 0 end) as 语文,`

`sum(case Course when '数学' then Grade else 0 end) as 数学,`

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

---

```
sum(case Course when '英语' then Grade else 0 end) as 英语
```

```
from TableA
```

```
group by Name
```

2)、横表转纵表的"SQL"示例

横表结构: TEST\_H2Z

ID	姓名	语文	数学	英语
1	张三	80	90	70
2	李四	90	85	95
3	王五	88	75	90

转换后的表结构:

ID	姓名	科目	成绩
1	张三	语文	80
2	张三	数学	90
3	张三	英语	70
4	李四	语文	90
5	李四	数学	80
6	李四	英语	99
7	王五	语文	85
8	王五	数学	96
9	王五	英语	88

横表转纵表 SQL 示例:

```
SELECT 姓名,'语文' AS 科目,语文 AS 成绩 FROM TEST_H2Z UNION ALL
SELECT 姓名,'数学' AS 科目,数学 AS 成绩 FROM TEST_H2Z UNION ALL
```

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

---

```
SELECT 姓名,'英语' AS 科目,英语 AS 成绩 FROM TEST_H2Z  
ORDER BY 姓名,科目 DESC;
```

117、删除姓名、年龄重复的记录（常考）

Id	name	age	salary
1	yzk	80	1000
2	yzk	80	2000
3	tom	20	20000
4	tom	20	20000
5	im	20	20000

//取得不重复的数据

```
select * from Persons  
where Id in  
(  
SELECT      MAX(Id) AS Expr1  
FROM        Persons  
GROUP BY Name, Age  
)
```

根据姓名、年龄分组，取出每组的 Id 最大值，然后将 Id 最大值之外的排除。

删除重复的数据：

```
delete from Persons  
where Id not in  
(  
SELECT      MAX(Id) AS Expr1  
FROM        Persons  
GROUP BY Name, Age  
)
```

118、介绍几个使用过的开源的项目

Lucene.net、NP01、JQuery、ASP.Net AJAX toolkit、Quartz.Net、JqueryUI、ASP.Net MVC。在 CodePlex、SourceForge 等网站上有更多的开源项目。

119、说出五个集合类

List、Dictionary、Set、Stack（先入后出）、Queue、Tree 等。

解答：HashSet 用于盛放不同的数据，相同的数据只保留一份

120 有一个 10 个数的数组，计算其中不重复数字的个数。{3,5,9,8,10,5,3}

工程化的非最优解答：用 HashSet

```
int[] values = { 3, 5, 9, 8, 10, 5, 3 };
HashSet<int> set = new HashSet<int>();
foreach (int i in values)
{
    set.Add(i);
}

foreach (int i in set)
{
    Console.WriteLine(i);
}
```

122、下面是一个由\*号组成的 4 行倒三角形图案。要求：1、输入倒三角形的行数，行数的取值 3-21 之间，对于非法的行数，要求抛出提示“非法行数！”；2、在屏幕上打印这个指定了行数的倒三角形。

```
*****
*****
***
*
```

123、一个文本文件含有如下内容：

4580616022644994|3000|赵涛

4580616022645017|6000|张屹

4580616022645090|3200|郑欣夏

上述文件每行为一个转账记录，第一列表示帐号，第二列表示金额，第三列表示开户人姓名。

创建一张数据库表（MS SQLServer 数据库，表名和字段名自拟），请将上述文件逐条插入此表中。

124、一个文本文件含有如下内容，分别表示姓名和成绩：

张三 90

李四 96

王五 78



提供用户一个控制台界面，允许用户输入要查询的姓名，输入姓名并且按回车以后，打印出此人的成绩，如果不输入姓名直接按回车则显示所有人的姓名以及成绩。（注意：不能使用数据库）

126、下面程序的执行结果是什么？

```
class Program
```

```
{  
  
    static void Main(string[] args)  
  
    {  
  
        Person p1 = new Person();//1:00  
  
        Console.WriteLine(DateTime.Now);  
  
        Thread.Sleep(1000 * 10);  
  
        Console.WriteLine(DateTime.Now);  
  
        Console.WriteLine(p1.Birthday);  
  
        //打印出来的是对象 New 出来的时间  
  
        //因为字段是对象被 new 出来的时候初始化的  
  
        Console.ReadKey();  
  
    }  
}
```

```
class Person
```

```
{
```

```
public DateTime BirthDay = DateTime.Now;  
  
}
```

通过在 `public DateTime BirthDay = DateTime.Now` 上设定断点来验证。

再难一点

```
class Person  
{  
  
    public static int A = 30;  
    static Person()//静态构造函数在static字段初始化完成后执行  
    { //静态构造函数只执行一次  
        A++;  
    }  
  
}
```

`public int B = A++;`字段的初始化赋值代码只是在new一个对象的时候执行，而不是每次用字段的时候都执行

```
public Person()  
{ B++; }  
}
```

程序的执行结果是什么？

```
Person p1 = new Person();  
Console.WriteLine(p1.B);  
Console.WriteLine(Person.A);  
Console.WriteLine(p1.B);  
Console.WriteLine(Person.A);  
  
Person p2 = new Person();  
Console.WriteLine(p2.B);  
Console.WriteLine(Person.A);
```

127、说一下 SQLServer 中索引的两种类型（常考）

参考：聚簇(或者叫做聚集，**cluster**)索引和非聚簇索引。

字典的拼音目录就是聚簇(**cluster**)索引，笔画目录就是非聚簇索引。这样查询“**G**到**M**的汉字”就非常快，而查询“**6**划到**8**划的字”则慢。

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

---

聚簇索引是一种特殊索引，它使数据按照索引的排序顺序存放表中。聚簇索引类似于字典，即所有词条在字典中都以字母顺序排列。聚簇索引实际上重组了表中的数据，所以你只能在表中建立一个聚簇索引。

当数据按值的范围查询时，聚簇索引就显得特别有用。因为所有 **SQLServer** 都必需先找到所查询范围的第一行，然后依次下去，直到该范围的最后一个值找到为止，并且保证了所有其他值也落在这个范围内。举一个例子，一个应用程序要查找首字母位于 **G** 和 **P** 之间的姓名列表，**SQLServer** 首先找到以字母 **G** 开头的名字，取出所有记录，直到找到以字母 **P** 开头的名字为止，这种方法使得查询过程非常高效。

进行大量数据改动的表不适宜用聚簇索引，因为 **SQLServer** 将不得不在表中维护行的次序。如果要索引的值极少，例如一个列包含的全都是 **1** 和 **0**，创建聚簇索引就不是个好主意。如果表经常由一个指定的列来排序，该列将是簇索引的最佳候选列。这是因为表中的数据已经为你排好序了。如果访问一个表并使用 **BETWEEN**、**<**、**>**、**>=** 或 **<=** 操作符来返回一个范围的值时，应该考虑使用聚簇索引。

128、沪江网网申题：

1) .能否脱离 VS 用类似 editplus 的工具手写代码？你觉得如何才能提高代码的效率和性能？

可以，使用 CSC.exe 来编译.cs 文件！

可以根据业务流程、业务数据的特点进行优化，比如可以采用缓存、索引、表驱动等来提升代码的效率和性能，同时不要进行无意义的代码优化，重点优化系统的性能瓶颈。

2) .如何提高页面的显示速度？假如一个页面的加载时间是 10.89s，你会用什么样的方式进行优化？

回答要大气！显得很牛！

提示：分而治之的思想。

首先要找出问题所在，是服务器端运行的速度慢还是服务器端到客户端的下载慢还是页面在浏览器的加载速度慢。

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

---

如果是服务器端运行速度慢，则找是数据库的原因还是算法的问题，如果是数据库的问题则尝试添加索引、优化 SQL 语句，如果是算法的问题，则优化算法。如果对于一些不经常改动的页面可以使用静态页技术！对于一些数据不需要及时更新的而且取数据的过程非常耗时可以使用缓存。页面中的内容可以按需加载（比如说可以像 verycd 网站的评论那样当用户需要看的时候再加载其内容），可以在图片需要显示的时候再进行加载。

如果是服务器端到客户端的下载慢则看是页面体积过于臃肿还是网络问题，如果是页面体积过于臃肿，则优化 HTML 代码，去掉无用的标签，压缩 JS、CSS，可以用 CSS Spirit 技术将多个图片放到一个图片中，减少向服务器的请求。如果是网络问题，则尝试在不同的网络、地区部署服务器，然后使用 CDN 技术加速访问。

如果是页面中的 JavaScript 运行复杂导致运行速度慢，则优化 JavaScript。

webkaka：测网速。

3) .在程序编码的时候，你会对 Sql 注入漏洞的防范采取什么样的措施？

尽量不要拼 SQL 语句！使用参数化查询或存储过程可以防止 SQL 注入攻击！在必须用 SQL 拼接的地方对用户输入的内容进行检查、过滤。

4) .你会采用什么样的策略和方法来实现系统缓存？

在 ASP.NET 中 缓存有 页面缓存，数据源缓存，和一些自己定义的缓存！

对于那些整个页面不经常变化的我们可以使用页面缓存，而对于那些执行非常耗时的 SQL 语句并且数据的及时性要求不高的我们可以使用数据源缓存。

对于页面缓存、数据源缓存等都不满足要求的情况下采用直接操作 `HttpRuntime.Cache` 来自定义缓存策略。如果需要多台 Web 服务器作为一个集群来承载网站的情况，则可以部署专门的分布式缓存服务器，比如 Memcached。

5) 网站想要实现文件防盗链的功能，说说你的解决方案。

读取 HTTP 报文头中的 `UrlReferrer` 在 `Application_BeginRequest` 中我们可以判断用户的请求是否来源于本网站。如果不是我们可以终止用户的请求（`Response.End();`）！

**不是贷款，零学费入学，不 3K 就业不给一分学费**  
**中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)**

---

6) .有这样一个功能需求，用户新注册的时候需要给他推荐 3 个好友，说说你的推荐算法

比如说类似豆瓣网我们可以根据用户兴趣来匹配，类似于人人网那些地区性比较比较强的网站我们可以根据用户的地区来匹配。

7) .如果给学英语的用户设计一个在线应用，你会选择什么作为切入点，简述你的想法和理由。

英语学习资料的下载、在线答疑、用户交流作为切入点。

8) .你眼中的沪江是怎样的？谈谈你对沪江的理解

我眼中的沪江是为那些爱好外语，需要学习外语人提供一个学校交流的平台！

9) .是否有以下相关开发经验？如有请简要说明开发情况（有可看项目网址也请提供）。

a、lucene 相关开发经验

b、分布式文件处理经验

c、流媒体相关开发经验

d、语音技术

e、高负载网站开发

f、手机相关开发

129、请编程遍历 WinForm 页面上所有 TextBox 控件并给它赋值为 string.Empty?

答：

```
foreach (System.Windows.Forms.Control control in this.Controls)
```

```
{
```

```
if (control is System.Windows.Forms.TextBox)
```

```
{  
System.Windows.Forms.TextBox tb = (System.Windows.Forms.TextBox)control ;  
tb.Text = String.Empty ;  
}  
}
```

132、

```
int i=2000;  
  
object o=i;  
  
i=2001;  
  
int j=(int)o;  
  
Console.WriteLine("i={0},o={1},j={2}",i,o,j);
```

输出的结果是：

134.一个数组：1,1,2,3,5,8,13,21...+m, 求第 30 位数是多少？用递归实现；(常考!!!)

写递归要确定两个：递归的终止条件；递归表达式。

解答：总结递归规律： $F(n)=F(n-2)+F(n-1)$  Fibonacci 数列

```
static int F(int n)
```

```
{
```

```
    if (n == 1)
```

```
    {
```

```
        return 1;
```

```
    }
```

```
    if (n == 2)
```

```
    {
```



```
        return 1;

    }

    return F(n - 2) + F(n - 1);

}
```

非递归(有 bug 吗? )，递归算法的缺点：测试大数据

```
int n = Convert.ToInt32(Console.ReadLine());
if(n<=0)
{Console.WriteLine("必须大于0");return;}
if(n==1) //时刻注意边界值!!!
{
    Console.WriteLine("1");
    return;
}
int[] data =new int[n];
data[0] = 1;
data[1] = 1;
for(int i=2;i<n;i++)
{
    data[i] = data[i-1]+data[i-2];
}

Console.WriteLine(data[n-1]);
```

上面程序的时间复杂度为  $O(n)$ ，空间复杂度为  $O(n)$

用 BigInteger 解决溢出的问题。

参考：

算法复杂度：**时间复杂度**：算法运行需要消耗的时间的数量级、**空间复杂度**：算法运行需要消耗的内存的数量级。

消耗的时间或者内存随着问题规模的扩大而成正比增加，就是  $O(n)$ 。

消耗的时间或者内存随着问题规模的扩大而不变，就是  $O(1)$ 。

消耗的时间或者内存随着问题规模的扩大而  $n*n$  增加，就是  $O(n*n)$

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 net.ITHeima.com

---

算法复杂度只考虑最差情况（从一个数组中找出第一个大于 10 的数，时间复杂度为  $O(n)$ ），并且算法复杂度忽略常量和低阶。把数组数一遍和数两遍的时间复杂度都是  $O(n)$ 。把长度为  $n$  的数组数  $n/2$  遍的时间复杂度还是  $O(n*n)$ 。

上课时把数组前后颠倒的算法的时间复杂度为  $O(n)$ ，空间复杂度为  $O(1)$ 。

136.ref 与 out 的区别？

137.表 A 字段 Id 为 numeric(18,0)，哪个 SQL 语句是错误的：

select \* from A where id=";

select \* from A where id='13';

select \* from A where id=null;

select \* from A where id=' 13';

138.在 SQLServer 中求当前时间与 2012-01-01 0: 0:0 相差的秒数？

140、做一个表格，三行三列，第一列，前两行合一；第二行，后两列合一。

141、下面的 HTML 代码

```
<style type="text/css">
```

```
    #Text1{ background-color:Red}
```

```
    .txt{ background-color:Green}
```

```
    input {background-color:Black}
```

```
</style>
```

```
<input id="Text1" type="text" class="txt" />
```

```
<input type="text" class="txt" />
```

两个文本框各是什么颜色？

参考资料：个人大于集体：元素单独设置的样式>class 命名样式>标签样式。样式 override 父级别的。

145、传智播客.net 培训精品就业班学员总结 SQL 题

表一: student\_info

学号	姓名	性别	出生年月	家庭住址	备注
0001	张三	男	1981-8-9	北京	NULL

表二: curriculum

课程编号	课程名称	学分
0001	计算机基础	2
0002	C语言	2

表三: grade

学号	课程编号	分数
0001	0001	80
0001	0002	90

题目:

条件查询:

- 在GRADE表中查找80-90份的学生学号和分数  
`select 学号,分数 from grade where 分数 between 80 and 90`
- 在GRADE 表中查找课程编号为003学生的平均分  
`select avg(分数) from grade where 课程编号='003'`
- 在GRADE 表中查询学习各门课程的人数  
`Select 课程编号,count(学号) as 人数 from grade group by 课程编号`
- 查询所有姓张的学生的学号和姓名  
`select 姓名,学号 from student_info where 姓名 like '张%'`

嵌套查询:

- 查询和学号'0001'的这位同学性别相同的所有同学的姓名和出生年月  
`select 姓名,出生年月 from student_info where 性别 in(select 性别 from student_info where sno='0001')`
- 查询所有选修课程编号为0002 和0003的学生的学号、姓名和性别  
`select 学号,姓名,性别 from student_info where 学号 in(select 学号 from grade where 课程编号='0002' and 学号 in(select 学号 from grade where 课程编号='0001'))`
- 查询出学号为0001的学生的分数比0002号学生最低分高的课程编号的课程编号和分数

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

```
select 课程编号, 分数 from grade where 学号='0001' and 分数>(select min(分数) from grade where 学号='0002')
```

多表查询:

- 1、查询分数在80-90分的学生的学号、姓名、分数

```
select student_info.学号,student_info.姓名,grade.分数 from student_info,grade where grade.分数 between 80 and 90
```

- 2、查询学习了'C语言'课程的学生学号、姓名和分数

```
select student_info.学号,student_info.姓名,grade.成绩from student_info,grade,curriculum where student_info.学号=grade.学号and grade.课程号=curriculum.课程号and curriculum.课程名='C语言'
```

- 3、查询所有学生的总成绩，要求列出学号、姓名、总成绩，没有选课的学生总成绩为空。

```
select grade.学号,student_info.姓名,sum(grade.成绩) as 总成绩 from student_info,grade where grade.学号=student_info.学号 group by grade.学号,student_info.姓名
```

题目、活期存款中，“储户”通过“存取款单”和“储蓄所”发生联系。假定储户包括：账号，姓名，电话，地址，存款额；“储蓄所”包括：储蓄所编号，名称，电话，地址（假定一个储户可以在不同得储蓄所存取款）

- 1、写出设计以上表格的语句（4 分）

- 2、创建一个触发器 TR1 完成下面内容：

当向“存取款单”表中插入数据时，如果存取标志 = 1 则应该更改储户表让存款额加上存取金额，如果存取标志 = 0 则应该更改储户表让存款额减去存取金额,如果余额不足显示余额不足错误。

```
CREATE TABLE CREATE TRIGGER tr1 on qukuan after insert AS BEGIN
```

```
declare @sid nvarchar(50)
declare @type int
declare @qian int
declare @yuer int
select @sid=sid,@type=[type],@m=m from inserted
select @yuer=yuer from cunkuan
if(@type=1)
begin
update cunkuan set yuer=yuer+@qian
end
else
begin
if(@yuer<@qian)
```

```
begin
    print '余额不足'
end
else
    begin
        update cunkuan set yuer=yuer-@qian
    end
end

END
GO
```

本题用到下面三个关系表：

CARD 借书卡: (CNO 卡号, NAME 姓名, CLASS 班级)

BOOKS 图书: (BNO 书号, BNAME 书名, AUTHOR 作者, PRICE 单价, QUANTITY 库存册数)

BORROW 借书记录: (CNO 借书卡号, BNO 书号, RDATE 还书日期)

备注：限定每人每种书只能借一本；库存册数随借书、还书而改变。

要求实现如下处理：

1. 写出自定义函数，要求输入借书卡号能得到该卡号所借书金额的总和

```
CREATE FUNCTION getSUM
(
    @CNO int
)
RETURNS int
AS
BEGIN
    declare @sum int
    select @sum=sum(price) from BOOKS where bno in (select bno from BORROW
    where cno=@CNO)

    return @sum
END
GO
```

2. 找出借书超过 5 本的读者,输出借书卡号及所借图书册数。

```
select CNO,count(BNO) as 借书数量 from BORROW group by CNO having
count(BNO)>3
```

3. 查询借阅了"水浒"一书的读者，输出姓名及班级。

```
select name,class from card where cno in( select cno from borrow where
```

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

---

```
bno in(  
select bno from BOOKS where bname='水浒'))
```

4. 查询过期未还图书，输出借阅者（卡号）、书号及还书日期。

```
select CNO,BNO,RDATE from borrow where getdate()>RDATE
```

5. 查询书名包括"网络"关键词的图书，输出书号、书名、作者。

```
select bno,bname,author from books where bname like '网络%'
```

6. 查询现有图书中价格最高的图书，输出书名及作者。

```
select bname,author from books where price in(select max(price) from  
books )
```

7. 查询当前借了"计算方法"但没有借"计算方法习题集"的读者，输出其借书卡号，并按卡号降序排序输出。

```
select cno from borrow where bno in (select bno from books where bname='  
计算方法') and cno not in ( select cno from borrow where bno in(select bno  
from books where bname='计算方法习题集')) order by cno desc
```

或

```
SELECT a.CNO  
FROM BORROW a,BOOKS b  
WHERE a.BNO=b.BNO AND b.BNAME=N'计算方法'  
AND NOT EXISTS(  
    SELECT * FROM BORROW aa,BOOKS bb  
    WHERE aa.BNO=bb.BNO  
        AND bb.BNAME=N'计算方法习题集'  
        AND aa.CNO=a.CNO)  
ORDER BY a.CNO DESC
```

8. 将"C01"班同学所借图书的还期都延长一周。

```
update borrow set rdate=dateadd(day,7,rdate) from BORROW where cno  
in(select cno from card where class='一班')
```

9. 从 BOOKS 表中删除当前无人借阅的图书记录。

```
DELETE A FROM BOOKS a  
WHERE NOT EXISTS(  
    SELECT * FROM BORROW  
    WHERE BNO=a.BNO)
```

10. 如果经常按书名查询图书信息，请建立合适的索引。



## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

(这个不确定对不 90%应该是对的 自己看了下书写的)

```
CREATE CLUSTERED INDEX IDX_BOOKS_BNAME ON BOOKS(BNAME)
```

11. 在 BORROW 表上建立一个触发器，完成如下功能：如果读者借阅的书名是"数据库技术及应用"，就将该读者的借阅记录保存在 BORROW\_SAVE 表中（注 BORROW\_SAVE 表结构同 BORROW 表）。

```
CREATE TRIGGER TR_SAVE ON BORROW
FOR INSERT,UPDATE
AS
IF @@ROWCOUNT>0
INSERT BORROW_SAVE SELECT i.*
FROM INSERTED i,BOOKS b
WHERE i.BNO=b.BNO
AND b.BNAME=N'数据库技术及应用'
```

12. 建立一个视图，显示"力 01"班学生的借书信息（只要求显示姓名和书名）。

```
CREATE VIEW V_VIEW
AS
select name,bname
from books,card,borrow
where borrow.cno=card.cno and borrow.bno=books.bno and class='一班'
```

13. 查询当前同时借有"计算方法"和"组合数学"两本书的读者，输出其借书卡号，并按卡号升序排序输出。

```
select a.cno from borrow a,borrow b
where a.cno=b.cno and
a.bno in(select bno from books where bname='计算方法') and
b.bno in(select bno from books where bname='组合数学')
order by a.cno desc
```

或

```
SELECT a.CNO
FROM BORROW a,BOOKS b
WHERE a.BNO=b.BNO
AND b.BNAME IN('计算方法','组合数学')
GROUP BY a.CNO
HAVING COUNT(*)=2
ORDER BY a.CNO DESC
```

14、用事务实现如下功能：一个借书卡号借走某书号的书，则该书的库存量减少 1，当某书的库存量不够 1 本的时候，该卡号不能借该书

```
alter PROCEDURE pro_jieshu
    @cno int,
    @bno int,
    @date datetime
AS
BEGIN

begin tran

declare @quantity int

select @quantity=quantity from books where bno=@bno
    insert into borrow values(@cno,@bno,@date)
    update books set quantity=@quantity-1 where bno=@bno
if(@quantity>0)
    begin
        commit tran
    end
else
    begin
        print '已无库存'
        rollback
    end
END
GO
```

15、用游标实现将书号为‘A001’的书本的价格提高 10 元

```
declare @bno int
declare @bname nvarchar(50)
declare @author nvarchar(50)
declare @price int
declare @quantity int

declare mycursor cursor for select * from books

open mycursor

fetch next from mycursor into @bno,@bname,@author,@price,@quantity

while(@@fetch_status=0)
    begin
        if(@bno=2)
```

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

---

```
begin
update books set price=@price+10 where current of mycursor
end
fetch next from mycursor into @bno,@bname,@author,@price,@quantity
end

close mycursor

deallocate mycursor
```

Student(S#,Sname,Sage,Ssex) 学生表

Course(C#,Cname,T#) 课程表

SC(S#,C#,score) 成绩表

Teacher(T#,Tname) 教师表

问题:

1、查询“001”课程比“002”课程成绩高的所有学生的学号;

```
select a.S# from (select s#,score from SC where C#='001') a,(select
s#,score
from SC where C#='002') b
where a.score>b.score and a.s#=b.s#;
```

2、查询平均成绩大于 60 分的同学的学号和平均成绩;

```
select S#,avg(score)
from sc
group by S# having avg(score) >60;
```

3、查询所有同学的学号、姓名、选课数、总成绩;

```
select Student.S#,Student.Sname,count(SC.C#),sum(score)
from Student left Outer join SC on Student.S#=SC.S#
group by Student.S#,Sname
```

4、查询姓“李”的老师的人数;

```
select count(distinct(Tname))
```

```
from Teacher
```

```
where Tname like '李%';
```

5、查询没学过“叶平”老师课的同学的学号、姓名；

```
select Student.S#,Student.Sname
```

```
from Student
```

```
where S# not in (select distinct( SC.S#) from SC,Course,Teacher where  
SC.C#=Course.C# and Teacher.T#=Course.T# and Teacher.Tname='叶平');
```

6、查询学过“001”并且也学过编号“002”课程的同学的学号、姓名；

```
select      Student.S#,Student.Sname      from      Student,SC      where
```

```
Student.S#=SC.S# and SC.C#='001' and exists( Select * from SC as SC_2 where  
SC_2.S#=SC.S# and SC_2.C#='002');
```

7、查询学过“叶平”老师所教的所有课的同学的学号、姓名；

```
select S#,Sname
```

```
from Student
```

```
where S# in (select S# from SC ,Course ,Teacher where SC.C#=Course.C#  
and Teacher.T#=Course.T# and Teacher.Tname='叶平' group by S# having  
count(SC.C#)=(select count(C#) from Course,Teacher where  
Teacher.T#=Course.T# and Tname='叶平'));
```

8、查询课程编号“002”的成绩比课程编号“001”课程低的所有同学的学号、姓名；

```
Select S#,Sname from (select Student.S#,Student.Sname,score ,(select  
score from SC SC_2 where SC_2.S#=Student.S# and SC_2.C#='002') score2  
from Student,SC where Student.S#=SC.S# and C#='001') S_2 where score2  
<score;
```

9、查询所有课程成绩小于 60 分的同学的学号、姓名；

```
select S#,Sname
```

```
from Student
```

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 net.ITHeima.com

```
where S# not in (select Student.S# from Student,SC where S.S#=SC.S# and  
score>60);
```

10、查询没有学全所有课的同学的学号、姓名；

```
select Student.S#,Student.Sname  
from Student,SC  
where Student.S#=SC.S# group by Student.S#,Student.Sname having  
count(C#) <(select count(C#) from Course);
```

11、查询至少有一门课与学号为“1001”的同学所学相同的同学的学号和姓名；

```
select S#,Sname from Student,SC where Student.S#=SC.S# and C# in  
select C# from SC where S#='1001';
```

12、查询至少学过学号为“001”同学所有一门课的其他同学学号和姓名；

```
select distinct SC.S#,Sname  
from Student,SC  
where Student.S#=SC.S# and C# in (select C# from SC where S#='001');
```

13、把“SC”表中“叶平”老师教的课的成绩都更改为此课程的平均成绩；

```
update SC set score=(select avg(SC_2.score)  
from SC SC_2  
where SC_2.C#=SC.C# ) from Course,Teacher where Course.C#=SC.C# and  
Course.T#=Teacher.T# and Teacher.Tname='叶平');
```

14、查询和“1002”号的同学学习的课程完全相同的其他同学学号和姓名；

```
select S# from SC where C# in (select C# from SC where S#='1002')  
group by S# having count(*)=(select count(*) from SC where  
S#='1002');
```

15、删除学习“叶平”老师课的 SC 表记录；

```
Delect SC  
from course ,Teacher  
where Course.C#=SC.C# and Course.T#= Teacher.T# and Tname='叶平';
```

16、向 SC 表中插入一些记录，这些记录要求符合以下条件：没有上过编号“003”课程的同学学号、2、

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 net.ITHeima.com

号课的平均成绩;

```
Insert SC select S#,'002',(Select avg(score)
from SC where C#='002') from Student where S# not in (Select S# from
SC where C#='002');
```

17、按平均成绩从高到低显示所有学生的“数据库”、“企业管理”、“英语”三门的课程成绩，按如下形式显示： 学生 ID,,数据库,企业管理,英语,有效课程数,有效平均分

```
SELECT S# as 学生 ID
, (SELECT score FROM SC WHERE SC.S#=t.S# AND C#='004') AS 数据库
, (SELECT score FROM SC WHERE SC.S#=t.S# AND C#='001') AS 企业管理
, (SELECT score FROM SC WHERE SC.S#=t.S# AND C#='006') AS 英语
, COUNT(*) AS 有效课程数, AVG(t.score) AS 平均分
FROM SC AS t
GROUP BY S#
ORDER BY avg(t.score)
```

18、查询各科成绩最高和最低的分：以如下形式显示：课程 ID，最高分，最低分

```
SELECT L.C# As 课程 ID,L.score AS 最高分,R.score AS 最低分
FROM SC L ,SC AS R
WHERE L.C# = R.C# and
      L.score = (SELECT MAX(IL.score)
                  FROM SC AS IL,Student AS IM
                  WHERE L.C# = IL.C# and IM.S#=IL.S#
                  GROUP BY IL.C#)
AND
      R.Score = (SELECT MIN(IR.score)
                  FROM SC AS IR
```

```
WHERE R.C# = IR.C#  
GROUP BY IR.C#  
);
```

19、按各科平均成绩从低到高和及格率的百分数从高到低顺序

```
SELECT t.C# AS 课程号 ,max(course.Cname)AS 课程  
名, isnull(AVG(score),0) AS 平均成绩  
,100 * SUM(CASE WHEN isnull(score,0)>=60 THEN 1 ELSE 0  
END)/COUNT(*) AS 及格百分数  
FROM SC T,Course  
where t.C#=course.C#  
GROUP BY t.C#  
ORDER BY 100 * SUM(CASE WHEN isnull(score,0)>=60 THEN 1 ELSE 0  
END)/COUNT(*) DESC
```

20、查询如下课程平均成绩和及格率的百分数(用"1行"显示)：企业管理(001)，  
马克思(002)，OO&UML(003)，数据库(004)

```
SELECT SUM(CASE WHEN C#='001' THEN score ELSE 0 END)/SUM(CASE C# WHEN  
'001' THEN 1 ELSE 0 END) AS 企业管理平均分  
,100 * SUM(CASE WHEN C#='001' AND score >= 60 THEN 1 ELSE 0  
END)/SUM(CASE WHEN C#='001' THEN 1 ELSE 0 END) AS 企业管理及格百分数  
,SUM(CASE WHEN C#='002' THEN score ELSE 0 END)/SUM(CASE C# WHEN  
'002' THEN 1 ELSE 0 END) AS 马克思平均分  
,100 * SUM(CASE WHEN C#='002' AND score >= 60 THEN 1 ELSE 0  
END)/SUM(CASE WHEN C#='002' THEN 1 ELSE 0 END) AS 马克思及格百分数  
,SUM(CASE WHEN C#='003' THEN score ELSE 0 END)/SUM(CASE C# WHEN  
'003' THEN 1 ELSE 0 END) AS UML 平均分  
,100 * SUM(CASE WHEN C#='003' AND score >= 60 THEN 1 ELSE 0  
END)/SUM(CASE WHEN C#='003' THEN 1 ELSE 0 END) AS UML 及格百分数  
,SUM(CASE WHEN C#='004' THEN score ELSE 0 END)/SUM(CASE C# WHEN  
'004' THEN 1 ELSE 0 END) AS 数据库平均分
```



不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 net.ITHeima.com

```
,100 * SUM(CASE WHEN C# = '004' AND score >= 60 THEN 1 ELSE 0  
END)/SUM(CASE WHEN C# = '004' THEN 1 ELSE 0 END) AS 数据库及格百分数  
FROM SC
```

21、查询不同老师所教不同课程平均分从高到低显示

```
SELECT max(Z.T#) AS 教师 ID,MAX(Z.Tname) AS 教师姓名,C.C# AS 课程 I  
D ,MAX(C.Cname) AS 课程名称,AVG(Score) AS 平均成绩  
FROM SC AS T,Course AS C ,Teacher AS Z  
where T.C#=C.C# and C.T#=Z.T#  
GROUP BY C.C#  
ORDER BY AVG(Score) DESC
```

22、查询如下课程成绩第 3 名到第 6 名的学生成绩单：企业管理（001），马  
克思（002），UML（003），数据库（004）

```
[学生 ID],[学生姓名],企业管理,马克思,UML,数据库,平均成绩  
SELECT DISTINCT top 3  
SC.S# As 学生学号,  
Student.Sname AS 学生姓名 ,  
T1.score AS 企业管理,  
T2.score AS 马克思,  
T3.score AS UML,  
T4.score AS 数据库,  
ISNULL(T1.score,0) + ISNULL(T2.score,0) + ISNULL(T3.score,0) +  
ISNULL(T4.score,0) as 总分  
FROM Student,SC LEFT JOIN SC AS T1  
ON SC.S# = T1.S# AND T1.C# = '001'  
LEFT JOIN SC AS T2  
ON SC.S# = T2.S# AND T2.C# = '002'
```

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

```
LEFT JOIN SC AS T3
ON SC.S# = T3.S# AND T3.C# = '003'
LEFT JOIN SC AS T4
ON SC.S# = T4.S# AND T4.C# = '004'
WHERE student.S#=SC.S# and
ISNULL(T1.score,0) + ISNULL(T2.score,0) + ISNULL(T3.score,0) +
ISNULL(T4.score,0)
NOT IN
(SELECT
DISTINCT
TOP 15 WITH TIES
ISNULL(T1.score,0) + ISNULL(T2.score,0) +
ISNULL(T3.score,0) + ISNULL(T4.score,0)
FROM sc
LEFT JOIN sc AS T1
ON sc.S# = T1.S# AND T1.C# = 'k1'
LEFT JOIN sc AS T2
ON sc.S# = T2.S# AND T2.C# = 'k2'
LEFT JOIN sc AS T3
ON sc.S# = T3.S# AND T3.C# = 'k3'
LEFT JOIN sc AS T4
ON sc.S# = T4.S# AND T4.C# = 'k4'
ORDER BY ISNULL(T1.score,0) + ISNULL(T2.score,0) +
ISNULL(T3.score,0) + ISNULL(T4.score,0) DESC);
```

23、统计列印各科成绩，各分数段人数：课程 ID，课程名称，[100-85]，[85-70]，[70-60]，[ <60]

```
SELECT SC.C# as 课程 ID, Cname as 课程名称
```

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

```
        ,SUM(CASE WHEN score BETWEEN 85 AND 100 THEN 1 ELSE 0 END) AS [100  
- 85]  
        ,SUM(CASE WHEN score BETWEEN 70 AND 85 THEN 1 ELSE 0 END) AS [85  
- 70]  
        ,SUM(CASE WHEN score BETWEEN 60 AND 70 THEN 1 ELSE 0 END) AS [70  
- 60]  
        ,SUM(CASE WHEN score < 60 THEN 1 ELSE 0 END) AS [60 -]  
FROM SC, Course  
where SC.C#=Course.C#  
GROUP BY SC.C#, Cname;
```

#### 24、查询学生平均成绩及其名次

```
SELECT 1+(SELECT COUNT( distinct 平均成绩)  
          FROM (SELECT S#,AVG(score) AS 平均成绩  
                FROM SC  
                GROUP BY S#  
                ) AS T1  
          WHERE 平均成绩 > T2.平均成绩) as 名次,  
S# as 学生学号,平均成绩  
FROM (SELECT S#,AVG(score) 平均成绩  
      FROM SC  
      GROUP BY S#  
      ) AS T2  
ORDER BY 平均成绩 desc;
```

#### 25、查询各科成绩前三名的记录:(不考虑成绩并列情况)

```
SELECT t1.S# as 学生 ID,t1.C# as 课程 ID,Score as 分数  
FROM SC t1  
WHERE score IN (SELECT TOP 3 score
```

```
FROM SC  
WHERE t1.C#= C#  
ORDER BY score DESC  
)  
ORDER BY t1.C#;
```

26、查询每门课程被选修的学生数

```
select c#,count(S#) from sc group by C#;
```

27、查询出只选修了一门课程的全部学生的学号和姓名

```
select SC.S#,Student.Sname,count(C#) AS 选课数  
from SC ,Student  
where SC.S#=Student.S# group by SC.S# ,Student.Sname having  
count(C#)=1;
```

28、查询男生、女生人数

```
Select count(Ssex) as 男生人数 from Student group by Ssex having  
Ssex='男';
```

```
Select count(Ssex) as 女生人数 from Student group by Ssex having  
Ssex='女';
```

29、查询姓“张”的学生名单

```
SELECT Sname FROM Student WHERE Sname like '张%';
```

30、查询同名同性学生名单，并统计同名人数

```
select Sname,count(*) from Student group by Sname having count(*)>1;;
```

31、1981 年出生的学生名单(注：Student 表中 Sage 列的类型是 datetime)

```
select Sname, CONVERT(char(11),DATEPART(year,Sage)) as age  
from student  
where CONVERT(char(11),DATEPART(year,Sage))='1981';
```

32、查询每门课程的平均成绩，结果按平均成绩升序排列，平均成绩相同时，按课程号降序排列

```
Select C#,Avg(score) from SC group by C# order by Avg(score),C# DESC ;
```

33、查询平均成绩大于 85 的所有学生的学号、姓名和平均成绩

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

```
select Sname,SC.S# ,avg(score)
from Student,SC
where Student.S#=SC.S# group by SC.S#,Sname having
avg(score)>85;
```

34、查询课程名称为“数据库”，且分数低于 60 的学生姓名和分数

```
Select Sname, isnull(score,0)
from Student,SC,Course
where SC.S#=Student.S# and SC.C#=Course.C# and Course.Cname='数据库'and score <60;
```

35、查询所有学生的选课情况;

```
SELECT SC.S#,SC.C#,Sname,Cname
FROM SC,Student,Course
where SC.S#=Student.S# and SC.C#=Course.C# ;
```

36、查询任何一门课程成绩在 70 分以上的姓名、课程名称和分数;

```
SELECT distinct student.S#,student.Sname,SC.C#,SC.score
FROM student,Sc
WHERE SC.score>=70 AND SC.S#=student.S#;
```

37、查询不及格的课程，并按课程号从大到小排列

```
select c# from sc where score <60 order by C# ;
```

38、查询课程编号为 003 且课程成绩在 80 分以上的学生的学号和姓名;

```
select SC.S#,Student.Sname from SC,Student where SC.S#=Student.S#
and Score>80 and C#='003';
```

39、求选了课程的学生人数

```
select count(*) from sc;
```

40、查询选修“叶平”老师所授课程的学生中，成绩最高的学生姓名及其成绩

```
select Student.Sname,score
from Student,SC,Course C,Teacher
```

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

```
where Student.S#=SC.S# and SC.C#=C.C# and C.T#=Teacher.T# and  
Teacher.Tname='叶平' and SC.score=(select max(score)from SC where  
C#=C.C# );
```

41、查询各个课程及相应的选修人数

```
select count(*) from sc group by C#;
```

42、查询不同课程成绩相同的学生的学号、课程号、学生成绩

```
select distinct A.S#,B.score from SC A ,SC B where A.Score=B.Score  
and A.C# <>B.C# ;
```

43、查询每门功成绩最好的前两名

```
SELECT t1.S# as 学生 ID,t1.C# as 课程 ID,Score as 分数  
FROM SC t1  
WHERE score IN (SELECT TOP 2 score  
FROM SC  
WHERE t1.C#= C#  
ORDER BY score DESC  
)  
ORDER BY t1.C#;
```

44、统计每门课程的学生选修人数（超过 10 人的课程才统计）。要求输出课程号和选修人数，查询结果按人数降序排列，查询结果按人数降序排列，若人数相同，按课程号升序排列

```
select C# as 课程号,count(*) as 人数  
from sc  
group by C#  
order by count(*) desc,c#
```

45、检索至少选修两门课程的学生学号

```
select S#  
from sc  
group by s#  
having count(*) > = 2
```

46、查询全部学生都选修的课程的课程号和课程名

```
select C#,Cname
from Course
where C# in (select c# from sc group by c#)
```

47、查询没学过“叶平”老师讲授的任一门课程的学生姓名

```
select Sname from Student where S# not in (select S# from
Course,Teacher,SC where Course.T#=Teacher.T# and SC.C#=course.C# and
Tname='叶平');
```

48、查询两门以上不及格课程的同学的学号及其平均成绩

```
select S#,avg(isnull(score,0)) from SC where S# in (select S# from
SC where score <60 group by S# having count(*)>2)group by S#;
```

49、检索“004”课程分数小于 60，按分数降序排列的同学学号

```
select S# from SC where C#='004'and score <60 order by score desc;
```

50、删除“002”同学的“001”课程的成绩

```
delete from Sc where S#='001'and C#='001';
```

14. 假定在建 BOOKS 表时没有定义主码，写出为 BOOKS 表追加定义主码的语句  
--实现代码：

```
ALTER TABLE BOOKS ADD PRIMARY KEY(BNO)
```

15.1 将 NAME 最大列宽增加到 10 个字符（假定原为 6 个字符）

--实现代码：

```
ALTER TABLE CARD ALTER COLUMN NAME varchar(10)
```

15.2 为该表增加 1 列 NAME（系名），可变长，最大 20 个字符

--实现代码：

```
ALTER TABLE CARD ADD 系名 varchar(20)
```

树的广度优先遍历

```
private void PrintNodeWidthFirst(IEnumerable<TreeNode> treenodes)
{
    //要遍历的下级节点
    List<TreeNode> listNodes = new List<TreeNode>();
    foreach (TreeNode childNode in treenodes)
    {
        //如果节点有子节点则将子节点加入listNodes以备稍后遍历
        if (childNode.Nodes.Count > 0)
        {

```



```
listNodes.AddRange(childNode.Nodes.OfType<TreeNode>());  
}  
textBox1.AppendText(childNode.Text + "\r\n");  
}  
//如果有待遍历的子节点  
if (listNodes.Count > 0)  
{  
    PrintNodeWidthFirst(listNodes);  
}  
}
```

锁的问题:

```
public void Test(int i)  
{  
    lock (this)  
    {  
        if (i > 10)  
        {  
            i--;  
            Test(i);  
        }  
    }  
}
```

执行 Test(10)是否会造成死锁?

答: 不会, 因为多线程才有锁的问题, 线程无法自己锁自己 (单线程必然是有先后执行顺序的, 没必要锁)  
问学网 2011

3、Math.Round(11.5)、Math.Round(11.3)、Math.Round(-11.5)、Math.Round(-11.3)的值各是多少?

答案: Round 四舍五入: Math.Round(-11.5)=-12, 因为是按照绝对值考虑。

扩展: Math.Ceiling(-11.6)=-11, 因为负数的天花板也在上面。

Math.Floor(-11.5)=-12

4、short s=1;s=s+1;有错吗? short s=1;s+=1;有错吗?

答案: 第一个有错, 因为1是int类型, int+short结果是int, 无法隐式转换, 改为s=(short)(s+1);就可以了。

第二个没错, 经过反编译发现编译器自动优化成s=(short)(s+1);

5、产生一个 int 数组, 长度为 100, 并向其中随机插入 1-100, 并且不能重复。(要求使用两种方法)

解答: Random 类是一个伪随机数算法, 原理:

r(n)=(sed+r(n-1))% 100;

```
class MyRand  
{  
    private int seed;
```

```
private int prevNumber = 10;
public MyRand(int seed)
{
    this.seed = seed;
}

public int Next()
{
    int newRand = (seed + prevNumber) % 100;
    prevNumber = newRand;
    return newRand;
}
```

生成的随机数是依赖于上一个的，所以叫“随机数序列”。，确定的种子产生确定的随机数序列。

为了避免每次生成的随机数序列都一样，所以一般采用当前系统运行的毫秒数

`Environment.TickCount`做种子。

这就明白为什么

```
//for (int i = 0; i < 10; i++)
//{
//    Random rand = new Random();
//    Console.WriteLine(rand.Next(100));
//}
```

是错的。

经过反编译，`Random`类的无参构造函数最终也是调用有参的，传递的就是当前毫秒数

`public Random() : this(Environment.TickCount)`

这就验证了

```
//for (int i = 0; i < 10; i++)
//{
//    Random rand = new Random();
//    Console.WriteLine(rand.Next(100));
//}
```

为什么一样，或者一半一样，是因为在同一毫秒内。

做法1:

```
List<int> list = new List<int>();
Random rand = new Random();
while (list.Count < 100)
{
    int number = rand.Next(1, 101); //>=1,<101
    if (!list.Contains(number)) //如果list中已经含有这个数，则不插入
    {
        list.Add(number);
    }
}
```

```
}

foreach (int i in list)
{
    Console.WriteLine(i);
}
```

做法2:

先把1-100这100个数按顺序放入数组arr，再重复100次下面的操作，生成两个介于  $\geq 0, < 100$  之间的随机数m、n，颠倒arr[m]和arr[n]的数。

```
int[] arr = new int[100];
//把100个数顺序放入
for (int i = 0; i < 100; i++)
{
    arr[i] = i + 1;
}

Random rand = new Random();
for (int i = 0; i < 100; i++)
{
    //随机生成两个位置
    int m = rand.Next(0, 100);
    int n = rand.Next(0, 100);

    //颠倒两个位置
    int temp = arr[m];
    arr[m] = arr[n];
    arr[n] = temp;
}
```

效率比第一种高的多，这个只需要N次，而上一种至少需要N\*N次

数据库3: 索引分为聚集索引和非聚集索引，聚集索引类似于拼音目录，非聚集索引类似于比划目录。一个表只能有一个聚集索引。索引都能提高查询效率、降低增、删、改效率，聚集索引更能提高按“范围”进行查询的效率（where age>20 and age<25）。建议在经常需要进行范围检索的字段上建聚集索引。

2、JavaScript中声明空数组。

[]

晨飞燕第一次

2、冒泡排序

```
for (int j = 0; j < nums.Length - 1; j++)
{
    for (int i = 0; i < nums.Length - 1 - j; i++)
    {
        if (nums[i] > nums[i + 1])
        {
            int temp = nums[i];
            nums[i] = nums[i + 1];
            nums[i + 1] = temp;
        }
    }
}
```

如果只是调用集合的Sort方法，是不满足人家要求的！一定要自己写！

4、进程内Session是存在内存中的，在IIS重启或者内存不足的情况下Session会丢失。所以使用进程外Session（StateServer）。见《ASP.net 3.5揭秘（第二卷）》102页。

5、select \* from t where id=(select max(id) from t)  
更简单的就是：select top 1 \* from t order by id desc

6、题有问题，应该增加一个主键字段

```
create table T_Persons(id bigint identity(1,1),name nvarchar(20),age
int)
```

```
insert into T_Persons(name,age) values('小明',20);
insert into T_Persons(name,age) values('小明',20);
insert into T_Persons(name,age) values('小黑',20);
insert into T_Persons(name,age) values('小明',20);
insert into T_Persons(name,age) values('小宏',20);
```

--取出name、age相同的id最小的数据

```
select min(id)
from T_Persons
group by name,age;--按照name和age分组
```

--把id等于最小值的取出来

```
select * from T_Persons where id in
(
select min(id)
from T_Persons
group by name,age--按照name和age分组
)
```

## 不是贷款，零学费入学，不 3K 就业不给一分学费 中关村黑马程序员.Net 训练营 net.ITHeima.com

---

答案：

--把不等于最小id的删掉

```
delete from T_Persons where id not in
(
select min(id)
from T_Persons
group by name,age--按照name和age分组
)
```

8、

//原理：a水壶灌水后(一旦a水壶无水后，则灌水)，倒入b水壶，

//如果b水壶内是3，则输出结果；

//若b水壶满，则b水壶倒掉，重新由a水壶倒水。

//假设a水壶需要灌水n次，b水壶需要灌水m次

```
int n=0; int m=0;
```

```
while (!(5 * n - 6 * m == 3))
```

```
{
```

```
n++;
```

```
m = 5 * n / 6;
```

```
}
```

```
Console.WriteLine("水壶a灌水{0}次,水壶b灌水{1}次",n,m);
```

9、别瞎回答，要重点突出自己的和别人不一样的地方。别犯以前同学最后一关说自己父母管自己很严的事情。

你为什么来我们公司？如果是知名公司，就说仰慕贵公司，（举例，看到你们网站发展速度快，还听说过关于你们技术团队、技术架构的介绍）；如果是不知名公司，就说喜欢这个行业。描述自己针对这个职位的优势

4、2+5+"8"得到的结果是什么？

解答：从左向右运算，int+string是string。所以是"78"

"8"+2+5是"825"，而"8"+"(2+5)"是“87”

5、

(1)

```
var x=1;
```

```
var y=0;
```

```
var z=0;
```

```
function add(n){n=n+1;return n;}
```

```
y=add(x);
```

```
function add(n){n=n+3;return n;}
```

```
z=add(x);
```

求y和z的值是多少？

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.itheima.com)

---

参考：JavaScript引擎会先解析所有的命名函数，再去从上向下执行js代码。所以第二个add覆盖了第一个add的定义。因此结果是4、4

(2) 传智播客增加一问

```
var x=1;
var y=0;
var z=0;
var add = function(n){n=n+1;return n;}
y=add(x);
add =function(n){n=n+3;return n;}
z=add(x);
求y和z的值是多少？
```

参考：var add = function(n){n=n+1;return n;}是把一个变量名add指向匿名函数。执行的时候add指向哪个函数，就执行哪个函数。从上向下执行。结果是2、4。所以写程序的时候最好用匿名函数。

命名函数的add就是函数的名字。而var add = function(n){n=n+1;return n;}则是创建一个匿名函数，并且把变量add指向它，这个函数还是匿名函数，并不会因为有一个变量指向它他就不是匿名函数了。

有的公司题量非常大，根本做不完，其实是在考察你哪方面最熟悉，那么挑最会做的先做。



不给1分学费即可入学！  
不3K就业不给学费！  
就业不做软件开发不给学费！  
就业后靠自己能力逐步给学费！

**中关村“黑马程序员”训练营**  
**.Net+WindowsPhone7培训**

<http://net.itheima.com>

欢迎访问黑马程序员.Net 训练营获取笔试面试宝典最新版：

不是贷款，零学费入学，不 3K 就业不给一分学费  
中关村黑马程序员.Net 训练营 [net.ITHeima.com](http://net.ITHeima.com)

---

<http://net.ITHeima.com>

学习 ASP.net 最好的参考资料就是以“不做只会拖控件的菜鸟”为课程宗旨的中关村黑马程序员.Net 训练营 <http://net.ITHeima.com> 的 ASP.Net 课程，可以下载他们的视频教程自学，下载地址：<http://edu.csdn.net/heima/aspnetvideo.html>