

ASP.NET 笔试题对 ASP.NET 程序员来说基本上是应聘的必备之物，那么有哪些问答是经常出现的呢？本文为您收集一些常见的 ASP.NET 笔试题目，供大家参考。

1. ASP.NET 笔试题目之 SQL 部分：

问：聚集索引和非聚集索引区别：

答：正文内容本身就是一种按照一定规则排列的目录称为“聚集索引”。ASP.NET 服务器目录纯粹是目录，正文纯粹是正文的排序方式称为“非聚集索引”

问：索引有什么意义？简单解释 簇索引(clustered)，及其优缺点

答：聚集索引通常可加快 UPDATE 和 DELETE 操作的速度，因为这两个操作需要读取大量的数据。创建或修改聚集索引可能要花很长时间，因为执行这两个操作时要在磁盘上对表的行进行重组。但是 INSERT 会变慢。可考虑将聚集索引用于包含数量有限的唯一值的列，如 state 列只包含 50 个唯一的州代码。使用下列运算符返回一个范围值的查询：BETWEEN、>、>=、< 和 <=。

问：隔离级别什么意思？有几种隔离级别，分别简单解释？

答：事务准备接受不一致数据的级别称为隔离级别。隔离级别是一个事务必须与其它事务进行隔离的程度。较低的隔离级别可以增加并发，但代价是降低数据的正确性。相反，较高的隔离级别可以确保数据的正确性，但可能对并发产生负面影响。应用程序要求的隔离级别确定了 SQL Server 使用的锁定行为。

SQL-92 定义了下列四种隔离级别，SQL Server 支持所有这些隔离级别：

◆未提交读（事务隔离的最低级别，仅可保证不读取物理损坏的数据）。

◆提交读（SQL Server 默认级别）。

◆可重复读。

◆可串行读（事务隔离的最高级别，事务之间完全隔离）。

如果事务在可串行读隔离级别上运行，则可以保证任何并发重叠事务均是串行的。

问：简述对事务的理解

答：事务必须运行于可重复读或更高的隔离级别以防止丢失更新。当两个事务检索相同的行，然后基于原检索的值对行进行更新时，会发生丢失更新。如果两个事务使用一个 **UPDATE** 语句更新行，并且不基于以前检索的值进行更新，则在默认的提交读隔离级别不会发生丢失更新。

2. ASP.NET 笔试题目之编程语言部分 (C/C++/C#)

问： 进程和线程

答： 一般进程被定义为一个程序运行中的实例。进程有两个部分组成

ASP.NET 服务器 1. [系统](#)内核对象（进程控制块）

ASP.NET 服务器 2. 运行空间（包跨运行所需的代码，数据和资源，如堆和栈）；

ASP.NET 服务器 ASP.NET 服务器 进程是不活跃的，它只提供了一个运行环境。

ASP.NET 服务器 ASP.NET 服务器 线程是进程中的运行分支，他在进程提供的环境中执行代码。一个进程至少有一个线程,否则就没有存在的必要。

ASP.NET 服务器 ASP.NET 服务器 线程包跨一个内核对象和一个栈.线程是 CPU 的调度单位。

ASP.NET 服务器 ASP.NET 服务器 线程是进程运行的最小单位，进程参与分配资源，如句柄，内存等，线程不参与分配，呵呵，进程间通过同步，互斥，共享保持通信！

问： 何时进行装箱、拆箱操作

答： 当引用类型与值类型之间转换时

问： 什么是 CLR

答： CLR(Common Language Runtime)普通语言运行时是整个.net framework 构建的基础，是实现.Net 跨平台，跨语言，代码安全等核心特性的关键

问： .Net 与 COM 的关系是什么

答: .Net 支持 COM 和 .Net 组件之间的互操作性, 实际上可通过 Runtime Callable Wrapper(RCW)来实现在 .Net 中获取 COM 组件, 通过 COM Callable Wrapper (CCW) 来实现在 COM 中获取 .Net 组件。

问: .Net 中的中间语言(IL)是什么

答: .Net 中的中间语言是指可以在普通语言运行时规范下, 执行时通过即时 (Just In Time)编译器转化为本地代码的, 与 CPU 独立的一族指令集合。它在 .Net 平台中是实现语言互操作的一个核心环节, 所有 .Net 平台的语言都要先被编译成中间语言(IL)。

3. ASP.NET 笔试题目之 WEB 开发部分

问: 在 ASP 文件头部写一句什么话将使 WEB SERVER 强制检查任何程序中使用到的变量必须先定义

答: Option explicit

问: 如果不希望某个错误影响到整个 ASP 页面的执行, 在头部写一句什么话可以做到

答: On error resume next

问: 希望 ASP 页面不进入缓存, 即每次执行都自动刷新, 写一句什么话可以做到

答: Response.Expires = -1

ASP.NET 服务器 Response.ExpiresAbsolute = Now() - 1

ASP.NET 服务器 Response.cachecontrol = "no-cache"