

教你轻松学会SQLServer记录轮班的技巧 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/259/2021\\_2022\\_\\_E6\\_95\\_99\\_E4\\_BD\\_A0\\_E8\\_BD\\_BB\\_E6\\_c98\\_259377.htm](https://www.100test.com/kao_ti2020/259/2021_2022__E6_95_99_E4_BD_A0_E8_BD_BB_E6_c98_259377.htm) 例：公司员工采取三个轮班制度：凌晨0：00到早上8：00为第一班，早上8：00到下午4：00为第二班，下午4：00到晚上12：00为第三班。员工使用电子时钟进行签名，这种电子签名可以自动将记录添加到SQL Server数据库中。但是，有时候，需要增加一个夜班；即使这个轮班发生在第二天，它仍然会被认为是第三班。你也许希望根据轮班对记录进行分组，但是在你的计划中会有两个非常令人头痛的问题：(1)不是所有的雇员都是按时到达的；(2)有时候员工工作的时间会比他们的一个轮班的时间长一些。为了能在你的列表中得到正确的分组，你必须假设所有的员工能在一个轮班的开始和结束时间之间的任意时间开始工作。使用CASE语句是一种解决办法。Listing A中的SQL语句告诉我们，在创建测试表没有错误的情况下，它是如何工作的。为了使程序代码完全准确的运行，你需要在时间的小窗口中得到每一个轮班的开始时间和下一个轮班的结束时间。在刚才的例子中，你可以用 ' 15:59:59.123 ' 这个值代替@StartTime，然后再重新运行代码。这时候，你必须在每一个轮班结束的分界点上增加999毫秒。解决方案 这个例子假设轮班时间从来没有改变过，但是如果改变了轮班时间该怎么办呢？你必须仿照这个例子并根据改变了的时间写出所有的代码。一个更好一点的主意就是，针对我们的轮班创建一个表来调用Shifts的开始时间和结束时间。在这个表中，你可以修改Case语句来查找StopTime列。 Listing B中包含了创

建Shifts表并向该表中增加记录的代码。代码在一个轮班中设置了时间后，这样使代码看上去既简单又灵活：  
DECLARE @StartTime VARCHAR(20) SET @StartTime = ' 22:59:59 '  
SELECT TOP 1 ShiftName FROM ShiftWork.Shifts WHERE Shifts.StartTime ORDERBY StartTime DESC 我之所以要将轮班记录进行降序排列，是为了避免第三个轮班的边界问题。我将用一些测试案例来论证我的结论，这些内容你可以在Listing C中看到。实际上，你可以将这个逻辑应用到TemeCards表的记录中去，而不需要创建另外的表，也就不需要在表中增加数据了。我同样使用一个简单的变量来论证我的逻辑。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)