

一条查询数据的SQL语句首先要被SQL Server 评估并生成对应的执行计划，最后才是根据执行计划来产生实际的查询操作并返回结果集给客户端，评估并生成执行计划是几件很消耗CPU的事情，因此SQL Server 并不会每次都去重新评估并生成执行计划，而且对于很多只是参数值不同的SQL来说他们的执行计划应该是一样的，比如对于如下的两个SQL来说就应该是这样的：

```
DECLARE @OrderID INT
SET @OrderID = 123
EXEC sp_executesql N'SELECT TOP 1 * FROM dbo.tblOrders WHERE OrderID = @OrderID', N'@OrderID'
GO

DECLARE @OrderID INT
SET @OrderID = 456
EXEC sp_executesql N'SELECT TOP 1 * FROM dbo.tblOrders WHERE OrderID = @OrderID', N'@OrderID'
GO
```

因此出于优化数据库系统性能考虑，一般的数据库系统都会对执行计划进行缓存，并设计一套机制来设法重用已经缓存的执行计划。

要想测试和观察SQL Server 如何生成和重用执行计划，可以使用下面的方法：

首先使用下面的语句来清空执行计划缓存：

```
DBCC FREEPROCCACHE
```

使用下面语句查看清空后的执行计划：

```
select * from sys.syscacheobjects
```

接着执行下面语句来使SQL Server生成执行计划：

```
DECLARE @OrderID INT
SET @OrderID = 123
EXEC sp_executesql N'SELECT TOP 1 * FROM dbo.tblOrders WHERE OrderID = @OrderID', N'@OrderID'
GO

DECLARE @OrderID INT
SET @OrderID = 456
EXEC sp_executesql N'SELECT TOP 1 * FROM dbo.tblOrders WHERE OrderID = @OrderID', N'@OrderID'
GO
```

最后使用下面语句观察执行计划的生成和重用机制

```
select * from sys.syscacheobjects where sql not like '%syscacheobjects%' and sql like '%tbl'
```

