

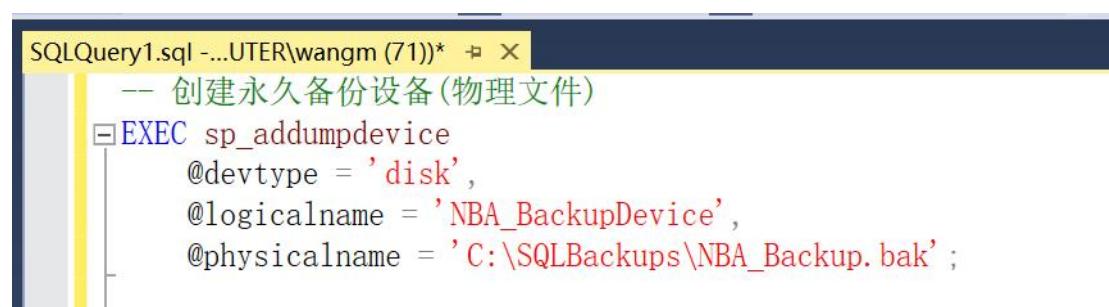
数据库实验-备份和恢复

1. 创建一个备份设备。

首先创建备份文件夹 SQLBackups:

xubei_ws.log	2023/7/13 11:06	文本文档	158 KB
SQLBackups	2025/5/21 18:54	文件夹	

然后创建备份设备，运行

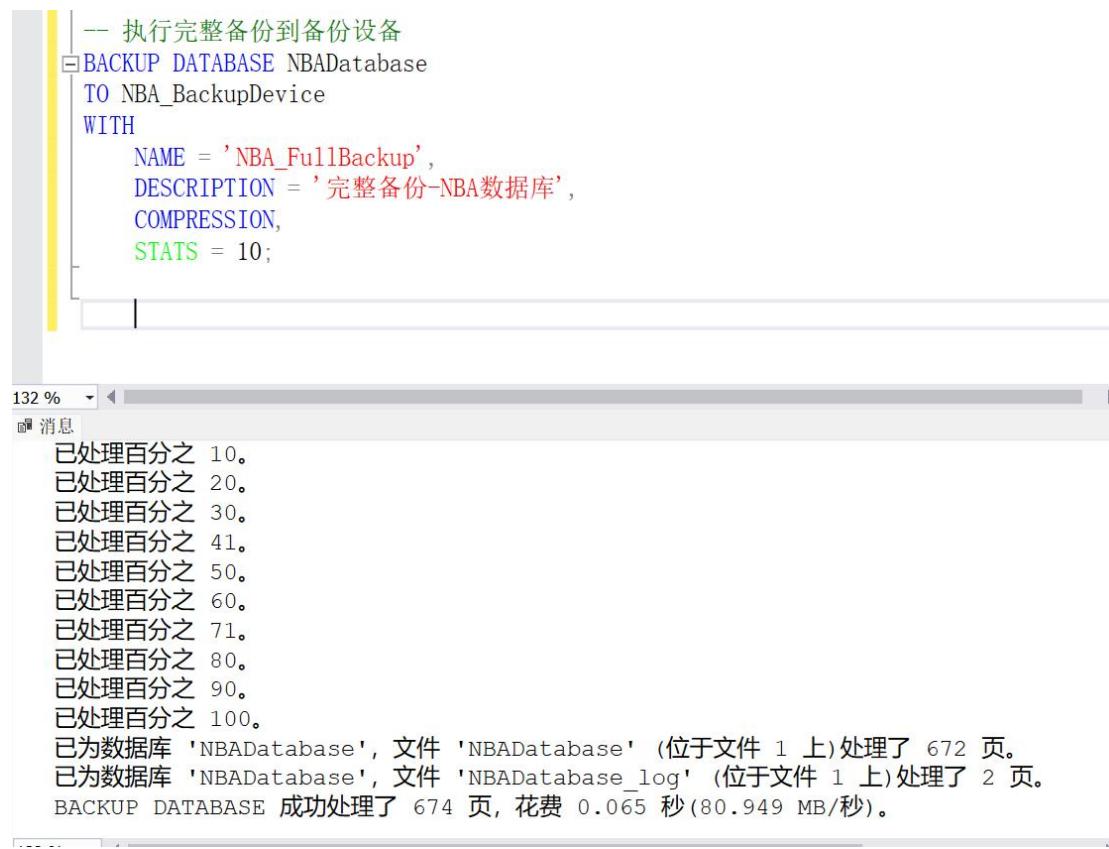


```
-- 创建永久备份设备(物理文件)
EXEC sp_addumpdevice
    @devtype = 'disk',
    @logicalname = 'NBA_BackupDevice',
    @physicalname = 'C:\SQLBackups\NBA_Backup.bak' ;
```

完成备份设备的创建；

2. 完整备份实验数据库到备份设备。

完整备份数据库



```
-- 执行完整备份到备份设备
BACKUP DATABASE NBADatabase
TO NBA_BackupDevice
WITH
    NAME = 'NBA_FullBackup',
    DESCRIPTION = '完整备份-NBA数据库',
    COMPRESSION,
    STATS = 10;
```

132 % 消息

```
已处理百分之 10。
已处理百分之 20。
已处理百分之 30。
已处理百分之 41。
已处理百分之 50。
已处理百分之 60。
已处理百分之 71。
已处理百分之 80。
已处理百分之 90。
已处理百分之 100。
已为数据库 'NBADatabase'，文件 'NBADatabase' (位于文件 1 上) 处理了 672 页。
已为数据库 'NBADatabase'，文件 'NBADatabase_log' (位于文件 1 上) 处理了 2 页。
BACKUP DATABASE 成功处理了 674 页，花费 0.065 秒(80.949 MB/秒)。
```

查看备份文件夹 SQLBackups

名称	修改日期	类型	大小
NBA_Backup.bak	2025/5/21 18:56	BAK 文件	980 KB

生成了 NBA_Backup.bak 文件；

3. 向实验数据库中某个表插入若干条记录。

首先记录当前状态：

-- 记录当前状态

```
SELECT COUNT(*) AS CurrentGames FROM Games;
```

然后向 Games 表中插入测试数据：

-- 插入测试数据

```
INSERT INTO Games (GameID, GameDate, HomeTeamID, AwayTeamID, HomeScore, AwayScore, Venue)
VALUES
    (100, '2023-11-01', 1, 2, 110, 105, '测试场馆A'),
    (101, '2023-11-02', 3, 4, 98, 102, '测试场馆B');
```

查询 Games 表检查插入完成：

-- 验证插入

```
SELECT * FROM Games WHERE GameID IN (100, 101);
```

32 %

结果 消息

	GameID	GameDate	HomeTeamID	AwayTeamID	HomeScore	AwayScore	Venue	Season
1	100	2023-11-01	1	2	110	105	测试场馆A	NULL
2	101	2023-11-02	3	4	98	102	测试场馆B	NULL

4. 备份数据库事务日志到备份设备，并查看日志格式。

备份数据库事务日志到备份设备：

```
-- 备份事务日志到同一设备  
BACKUP LOG NBADatabase  
TO NBA_BackupDevice  
WITH  
    NAME = 'NBA_LogBackup',  
    DESCRIPTION = '日志备份-测试数据插入后',  
    STATS = 10;
```

查看日志格式:

	BackupName	BackupDescription	BackupType	ExpirationDate	Compressed	Position	DeviceType	UserName	ServerName	DatabaseName	DatabaseVersion	DatabaseCreationDate	BackupSize	FirstLSN	LastLSN	Ch
1	NBA_FullBackup	完整备份-NBA数据库	1	NULL	1	1	102	VMY-COMPUTER\wangz	VMY-COMPUTER	NBADatabase	957	2025-05-04 11:08:30.000	5596136	40000000333200001	40000000337600001	40K
2	NBA_LogBackup	日志备份-测试数据插入后	2	NULL	1	2	102	VMY-COMPUTER\wangz	VMY-COMPUTER	NBADatabase	957	2025-05-04 11:08:30.000	143360	40000000335200001	40000000356800001	40K

5. 利用第2步所得的完整备份，恢复到插入记录前的状态。

先结束所有连接到 NBA 数据库的会话:

```
-- 首先结束所有连接到NBA数据库的会话  
USE master;  
GO  
  
ALTER DATABASE NBADatabase SET SINGLE_USER WITH ROLLBACK IMMEDIATE;
```

进行完整恢复:

```
-- 执行完整恢复(会覆盖现有数据库)  
RESTORE DATABASE NBADatabase  
FROM NBA_BackupDevice  
WITH  
    FILE = 1, -- 文件位置可通过RESTORE HEADERONLY查看  
    REPLACE,  
    RECOVERY; -- 完全恢复模式
```

切换回多用户模式并验证数据:

```
-- 切换回多用户模式  
ALTER DATABASE NBADatabase SET MULTI_USER;
```

```
-- 验证数据(应该看不到测试数据)
SELECT * FROM Games;
```

132 %

	GameID	GameDate	HomeTeamID	AwayTeamID	HomeScore	AwayScore	Venue	Season
1	1	2023-10-24	1	2	123	120	Crypto.com Arena	2023-24
2	2	2023-10-25	3	5	110	115	TD Garden	2023-24
3	3	2023-10-26	2	4	118	112	Chase Center	2023-24
4	4	2023-10-27	1	3	105	108	Crypto.com Arena	2023-24
5	5	2023-11-01	1	2	100	98	测试场馆	2023-24

果然 Games 表中 GameID 为 100 和 101 的表项消失了；

6. 利用第 4 步所得的事务日志，恢复到插入记录后的状态。

先再次恢复到完整备份状态：

```
-- 再次恢复到完整备份状态(准备应用日志)
USE master;
GO

ALTER DATABASE NBADatabase SET SINGLE_USER WITH ROLLBACK IMMEDIATE;
```

恢复完整备份：

```
-- 先恢复完整备份(NORECOVERY状态)
RESTORE DATABASE NBADatabase
FROM NBA_BackupDevice
WITH
FILE = 1,
REPLACE,
NORECOVERY; -- 关键区别
```

再应用事务日志备份：

```
-- 然后应用事务日志备份
RESTORE LOG NBADatabase
FROM NBA_BackupDevice
WITH
FILE = 2, -- 日志备份的位置
RECOVERY; -- 完成恢复
```

最后切换回多用户模式查询 Games 表验证恢复备份完成

```
-- 切换回多用户模式
```

```
ALTER DATABASE NBADatabase SET MULTI_USER;
```

The screenshot shows a SQL Server Management Studio window with three tabs at the top: 'SQLQuery3.sql -...UTER\wangm (74)*' (selected), 'SQLQuery2.sql -...UTER\wangm (79)*', and 'SQLQuery1.sql -...UTER\wangm (71)*'. The main area displays the results of a query:

```
-- 验证数据(应该能看到测试数据)
SELECT * FROM Games
```

The results grid shows the following data:

	GameID	GameDate	HomeTeamID	AwayTeamID	HomeScore	AwayScore	Venue	Season
1	1	2023-10-24	1	2	123	120	Crypto.com Arena	2023-24
2	2	2023-10-25	3	5	110	115	TD Garden	2023-24
3	3	2023-10-26	2	4	118	112	Chase Center	2023-24
4	4	2023-10-27	1	3	105	108	Crypto.com Arena	2023-24
5	5	2023-11-01	1	2	100	98	测试场馆	2023-24
6	100	2023-11-01	1	2	110	105	测试场馆A	NULL
7	101	2023-11-02	3	4	98	102	测试场馆B	NULL

果然又能查询到 GameID 为 100 和 101 的表项了；

最后清理测试数据：

```
-- 清理测试数据(实验完成后)
```

```
DELETE FROM Games WHERE GameID IN (100, 101);
```

完成备份和恢复实验。