

# 数据库编程简介

# Transact-SQL

- 标准SQL的非过程特性
- 每种数据库产品会对标准SQL进行扩充以便支持某些结构化程序的特征
- SQL Server ===》 Transact-SQL
- Oracle===》 PL/SQL
- Etc.



# Transact-SQL元素

- 标识符
- 数据类型
- 常量
- 函数
- 表达式
- 注释
- 保留关键字

# Transact-SQL过程的类型

- 批处理
  - 一条或多条语句序列 每次执行前编译
- 存储过程
  - 一组SQL语句，一次编译 多次执行
  - 可以提高性能
- 触发器
  - 特殊的存储过程，不由用户调用，通过事件激活
- 脚本
  - 存储在文件中的一系列语句，可以包括多个批处理



# Transact-SQL变量和参数

```
USE AdventureWorks;
GO
DECLARE @EmpIDVar int; --定义变量@EmpIDVar, 数据类型为int
SET @EmpIDVar = 1234;   --设置变量的值
SELECT *
FROM HumanResources.Employee
WHERE EmployeeID = @EmpIDVar;
/*使用变量作为查询条件的值*/
```

```
USE AdventureWorks;
GO
/* 存储过程ParmSample的参数
@EmpIDParm */
CREATE PROCEDURE ParmSample
@EmpIDParm int AS
SELECT EmployeeID, Title
FROM HumanResources.Employee
WHERE EmployeeID =
@EmpIDParm
GO
/* 批处理中通过参数传递值给存储过程
*/
EXEC ParmSample @EmpIDParm =
109
GO
```

# Transact-SQL流程控制

BEGIN...END	BEGIN 和 END 语句用于将多个 Transact-SQL 语句组合为一个逻辑块。
GOTO	GOTO 语句使 Transact-SQL 批处理的执行跳至标签。
IF...ELSE	IF 语句用于条件的测试。得到的控制流取决于是否指定了可选的 ELSE 语句。
RETURN	RETURN 语句无条件终止查询、存储过程或批处理。
WAITFOR	WAITFOR 语句挂起批处理、存储过程或事务的执行，直到发生以下情况： ① 已超过指定的时间间隔。 ② 到达一天中指定的时间。 ③ 指定的 RECEIVE 语句至少修改一行或并将其返回到 Service Broker 队列。
WHILE	只要指定的条件为 True 时，WHILE 语句就会重复语句或语句块。
CONTINUE	重新开始 WHILE 循环。
BREAK	终止 WHILE 循环。
CASE	CASE 函数用于计算多个条件并为每个条件返回单个值。



# Transact-SQL中的游标

- 原理和嵌入式SQL相同
  - 声明
  - 打开
  - 移动（提取）
  - 关闭

## Transact-SQL存储过程

- 存储过程：由Transact-SQL语句书写的过程，经编译和优化后存储在数据库服务器中，使用时只要调用即可。
- 存储过程的优点：
  1. 运行效率高
  2. 降低了客户机和服务器之间的通信量
  3. 方便实施企业规则



# 存储过程分类

- 用户定义存储过程
- 系统存储过程

## 存储过程（续）

- 存储过程的用户接口：
  1. 创建存储过程
  2. 执行存储过程
  3. 删除存储过程



## 二、存储过程的用户接口

- 1. 创建存储过程:

CREATE Procedure 过程名 ( [参数1, 参数2, ...] ) AS  
<SQL块>;

- 过程名: 数据库服务器合法的对象标识
- 参数列表: 用名字来标识调用时给出的参数值, 必须指定值的数据类型。参数也可以定义输入参数、输出参数或输入/输出参数。默认为输入参数。
- 过程体: 是一个<SQL块>。包括声明部分和可执行语句部分

## 存储过程的用户接口（续）

[例] 利用存储过程来实现下面的应用: 从一个账户转指定数额的款项到另一个账户中。

```
CREATE PROCEDURE TRANSFER(inAccount INT,  outAccount INT,  
amount FLOAT)  
AS DECLARE  
    totalDeposit FLOAT;  
BEGIN                                /* 检查转出账户的余额 */  
    SELECT total INTO totalDeposit  
    FROM ACCOUNT WHERE ACCOUNTNUM=outAccount;  
    IF totalDeposit IS NULL THEN /* 账户不存在或账户中没有存款 */  
        ROLLBACK;  
        RETURN;  
    END IF;
```



## 存储过程的用户接口（续）

```
IF totalDeposit < amount THEN      /* 账户账户存款不足 */  
    ROLLBACK;  
    RETURN;  
END IF;  
UPDATE account SET total=total-amount  
WHERE ACCOUNTNUM=outAccount;  
                                     /* 修改转出账户，减去转出额 */  
UPDATE account SET total=total + amount WHERE  
ACCOUNTNUM=inAccount;  
                                     /* 修改转入账户，增加转出额 */  
COMMIT;                             /* 提交转账事务 */  
END;
```

## 存储过程的用户接口（续）

- 2. 执行存储过程：

EXECUTE Procedure 过程名 [参数1, 参数2, ...] ;



## 存储过程的用户接口（续）

### 3. 删除存储过程

**DROP PROCEDURE** 过程名;

# Transact-SQL用户定义函数

- 函数：接受参数 执行处理 返回结果的程序
- 返回值的种类
  - 单个标量值-标量函数-通过RETURN语句返回单值
  - 结果集-表值函数
    - 内联表值函数-单查询语句定义 可以带有参数 可以看成带参数视图
    - 多语句表值函数-可以同多语句完成结果集的计算



# Transact-SQL触发器

- 触发器是数据库服务器中发生事件时自动执行的存储过程
- DML触发器-针对数据的更新操作
- DDL触发器
- 登陆触发器

# Transact-SQL触发器

- 注意和标准SQL的区别
- **INSTEAD OF** 触发器
  - 先于更新操作执行
  - 替代原有更新操作的执行
- **AFTER** 触发器
  - 更新事件发生后执行
  - 只能定义在表上



# 小节

