SQL

SQL语句的分类

1. 数据查询语句(DQL: Data Query Language)
   SELECT，搭配使用WHERE，ORDER BY，GROUP BY，HAVING
2. 数据操作语句(DML: Data Manipulation Language)
   INSERT，UPDATE，DELETE
3. 事务处理语句(TPL)
   BEGIN TRANSACTION，COMMIT，ROLLBACK
4. 数据控制语句(DCL)
   GRANT，REVOKE
5. 数据定义语句(DDL)
   CREATE，DROP
6. 指针控制语句(CCL)
   DECLARE CURSOR，FETCH INTO，UPDATE WHERE CURRENT

SQL数据类型

— 字符型(varchar相对char较优)
— 文本型(慎用，至少分配2k的空间，即便是空值)
— 数值型(小数numeric，钱数money)
— 逻辑型(建表时创建，建表后不能添加这样的字段)
— 日期型

一些重要的SQL命令

— SELECT
— UPDATE
— DELETE
— INSERT INTO
— CREATE DATABASE
— ALTER DATABASE
— CREATE TABLE
— ALTER TABLE
— DROP TABLE
— CREATE INDEX
— DROP INDEX

SQL 语句快速参考

关键字	语句格式
AND / OR	SELECT column_name(s) FROM table_name WHERE condition AND|OR condition
ALTER TABLE	ALTER TABLE table_name ADD column_name datatype or ALTER TABLE table_name DROP COLUMN column_name
AS(alias)	SELECT column_name AS column_alias FROM table_name or SELECT column_name FROM table_name AS table_alias
BETWEEN	SELECT column_name(s) FROM table_name WHERE column_name BETWEEN value1 AND value2
CREARE DATABASE	CREATE DATABASE database_name
CREATE TABLE	CREATE TABLE table_name(column_name1 data_type,column_name2 data_type,column_name3 data_tyoe,...)
CREATE INDEX	CREATE INDEX index_name ON table_name(column_name) or CREATE UNIQUE INDEX index_name ON table_name(column_name)
CREATE VIEW	CREATE VIEW view_name AS SELECT column_name(s) FROM table_name WHERE condition
DELETE	DELETE FROM table_name WHERE some_column=some_value or DELETE FROM table_name (Note:Deletes the entire table!!) or DEL
DROP DATABASE	DROP DATABASE database_name
DROP INDEX	DROP INDEX table_name.index_name (SQL Server) / DROP INDEX index_name On table_name (MS Access) / DROP INDEX index_name ALTER TABLE table_name DROP INDEX index_name (MySQL)
DROP TABLE	DROP TABLE table_name
GROUP BY	SELECT column_name,aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_na
HAVING	SELECT column_name,aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_na
IN	SELECT column_name(s) FROM table_name WHERE column_name IN (value1,value2,...)
INSERT INTO	INSERT INTO table_name VALUES(value1,value2,value3,...) or INSERT INTO table_name(column1,column2,column3,...) VALUES
INNER JOIN	SELECT column_name(s) FROM table_name1 INNER JOIN table_name2 ON table_name1.column_name=table_name2.column_name
LEFT JION	SELECT column_names(s) FROM table_name1 LEFT JOIN table_name2 ON table_name1.column_name=table_name2.column_name
RIGHT JOIN	SELECT column_name(s) FROM table_name1 RIGHT JOIN table_name2 ON table_name1.column=table_name2.column_name
FULL JOIN	SELECT column_name(s) FROM table_name1 FULL JOIN table2 ON table_name1.column_name=table_name2.column_name
LIKE	SELECT column_name(s) FROM table_name WHERE column_name LIKE pattern
ORDER BY	SELECT column_name(s) FROM table_name ORDER BY column_name [ASC/DESC]
SELECT	SELECT column_name(s) FROM table_name
SELECT *	SELECT * FROM table_name
SELECT INTO	SELECT * INTO new_table_name [IN externldatabase] FROM old_table_name or SELECT column_name(s) INTO new_table_name {I

INTO	
SELECT TOP	SELECT TOP number/percent column_name(s) FROM table_name
TRUNCATE TABLE	TRUNCATE TABLE table_name
UNION	SELECT column_name(s) FROM table_name1 UNION SELECT column(s) FROM table_name2
UNION ALL	SELECT column_name(s) FROM table_name1 UNION ALL SELECT column_name(s) FROM table_name2
UPDATE	UPDATE table_name SET column1=value1,column2=value2,... WHERE some_column=some_value
WHERE	SELECT column_name(s) FROM table_name WHERE column_name operator value

```
AND/OR SELECT column_name(s) FROM table_name WHERE condition AND|OR condition
```