SQL Server QUICK GUIDE
FOR THE BEGINNING USER

BASIC SYNTAX AND EXAMPLES

Note: There are many variations and extensions to the syntax and examples provided below. See SQL Server Books Online for complete syntax diagrams and usage examples.


DDL (Data Definition) WITH TABLES...

CREATE TABLE table_name (column definition) [table constraint] (
[,...n])

CREATE TABLE MyCustomers (
  CustID INT IDENTITY (100, 1) PRIMARY KEY,
  CompanyName NVARCHAR2(50)
)

CREATE TABLE MyOrders (OrderID int, CustID int REFERENCES MyCustomers(CustID))

ALTER TABLE table_name ALTER COLUMN column_name ADD column_definition | DROP column_definition | DROP column constraint

ALTER TABLE MyCustomers ALTER COLUMN CustID IDENTITY(200, 2)

ALTER TABLE Employee DROP COLUMN bday

ALTER TABLE MyCustomers ALTER COLUMN CompanyName DROP DEFAULT

DROP TABLE names_cpy

DML (Data Manipulation) WITH TABLES...

SELECT is different than other DML statements in that it does not actually change/manipulate the data by itself. It is sometimes used with other commands to carry out DML using data retrieved from elsewhere in the database. Following is the basic query-only syntax.

SELECT col1, col2, ...colx | * FROM table_name
WHERE colx = expr | [ORDER BY colx];

SELECT fname, lname FROM staff
WHERE phone IS NOT NULL
ORDER BY lname;

True Data Manipulation...

INSERT INTO staff (fname, lname, ssn) VALUES ('Chris', 'Plum', 318675309);

INSERT INTO staff SELECT * FROM new_hires;

DELETE FROM staff WHERE lname = 'Smith';

TRUNCATE TABLE Customers;

BULK INSERT AdventureWorks.Sales.SalesOrderDetail
FROM 'f:\orders\lineitem.tbl'
WITH (FIELDTERMINATOR = \',\', ROWTERMINATOR = \n);

DCL (Data Control) ... and an intro to roles and permissions.

Permissions are required to access the database. First a login must be created on the instance level. Then that login is used to create a database user. Then permissions must be assigned to the database. This is often done with the use of roles to simplify security administration. Roles are broken down into two categories: Server level roles and database level roles.

Fixed Server Roles

bulkadmin Members can run the BULK INSERT statement.
dbcreator Members can create, alter, drop, and restore any database.
diskadmin Members can manage disk files.
processadmin Members can terminate processes that are running in an instance of SQL Server.
securityadmin Members can manage logins and their properties.
serviceadmin Members can change server-wide configuration options and shut down the server.
setupadmin Members can add remove linked servers, and also execute some system stored procedures.
sysadmin Members can perform any activity in the server.

Fixed Database Roles

db_accessadmin Granted: ALTER ANY USER, CREATE SCHEMA

Granted with GRANT option: CONNECT

$db_accessadmin Granted with GRANT option: CONNECT

$db_backupoperator Granted: BACKUP DATABASE, BACKUP LOG, CHECKPOINT

$db_datareader Granted: SELECT

$db_datawriter Granted: DELETE, INSERT, UPDATE

$db_owner Granted with GRANT option: CONTROL

Granted: VIEW ANY DATABASE

$db_monitor Granted: VIEW most recent status in Database Mirroring Monitor

Important: The db_monitor fixed database role is created in the mdb database when the first database is registered in Database Mirroring Monitor. The new db_monitor role has no members until a system administrator assigns users to the role. Granted: VIEW ANY DATABASE

$dbdenydatareader Denied: SELECT

$dbdenydatareader Denied: DELETE, INSERT, UPDATE

$db_securityadmin Granted: ALTER ANY APPLICATION ROLE, ALTER ANY ROLE, CREATE SCHEMA, VIEW DEFINITION

Granted: VIEW ANY DATABASE
Perpetual Technologies, Inc. (PTI) provides mission-critical database and information systems support to commercial and government enterprises worldwide. Focused on improving performance and lowering costs, our subject-matter experts plan, design, develop, deploy, and manage SQL Server database environments to your exact specifications.

There are many good online references for SQL statement execution. This document only bullet-points some of the very basic commands. It is a very powerful language that, when exploited, will enable you to produce very complex reports.

Admin SQL

| GRANT permission [ , ...n ] ON <object> TO <database_principal> [WITH GRANT OPTION] | GRANT IMPERSONATE ON USER::NamithaL TO AccountsPayable1; GO |

BACKUP DATABASE Customers TO DISK = N'C:\DataBackup\Customers_db_200706041200.BAK'

RESTORE DATABASE Customers FROM DISK = N'C:\DataBackup\Customers_db_200706041200.BAK'

BACKUP LOG Customers TO DISK = N'C:\LogBackup\Customers_db_200706041200.TRN'

KILL 55;

(Find out who has current sessions on instance prior to shutting down)
Exec sp_who; GO
SHUTDOWN; GO

CONCLUSION

SQL Server Quick Reference Guide FOR THE BEGINNING USER

5649 Lee Road
Indianapolis, IN 46216
800-538-0453
www.PTI.net

ADVANCING THE DATA-DRIVEN WORLD