Cheatography

T-SQL Cheat Sheet

by giangpdh via cheatography.com/137135/cs/28755/

Stored Procedure

```
CREATE OR ALTER PROCEDURE
dbo.usp_Name(@param)
AS
BEGIN
  SET NOCOUNT ON;
  SET XACT_ABORT ON;
  BEGIN TRANSACTION
  COMMIT
  SET XACT_ABORT OFF;
  SET NOCOUNT OFF;
END
```

Add not null column to existing table

```
IF NOT EXISTS(SELECT 1 FROM
sys.columns WHERE Name =
N'MyNewColumn' AND Object_ID =
Object_ID(N'[dbo].[MyTable]'))
BEGIN
ALTER TABLE MyTable
        ADD MyNewColumn INT NOT
NULL DEFAULT 0
END
```

drop column

```
ALTER TABLE table_name
DROP COLUMN column_name;
```

Detect duplicate

```
SELECT *, ROW_NUMBER()
OVER(PARTITION BY TenantId ORDER
BY Id) from dbo.[Test] WHERE
[Name] is null
```

Check Exists

```
IF [NOT] EXISTS ( SELECT 1 FROM
MyTable WHERE ... )
     <do smth>
IF NOT EXISTS ( SELECT 1 FROM
Users WHERE FirstName = 'John'
AND LastName = 'Smith' )
BEGIN
    INSERT INTO Users (First-
Name, LastName) VALUES ('John',
'Smith')
END
```

Drop temp table

```
DROP TABLE IF EXISTS #Results
```

If else

```
IF Boolean_expression
     { sql_statement | statem-
ent_block }
[ ELSE
     { sql_statement | statem-
ent_block } ]
```

Merge

```
MERGE <target_table> AS tgt
USING <table_source> AS src
ON <merge_search_condition>
WHEN MATCHED THEN <merge_match-
ed>
WHEN NOT MATCHED [ BY TARGET ]
THEN <merge_not_matched>
WHEN NOT MATCHED BY SOURCE THEN
<merge_matched>
MERGE Production.UnitMeasure AS
tgt
USING (SELECT @UnitMeasureCode,
@Name) as src (UnitMeasureCode,
Name)
ON (tgt.UnitMeasureCode =
src.UnitMeasureCode)
WHEN MATCHED THEN
UPDATE SET Name = src.Name
WHEN NOT MATCHED THEN
INSERT (UnitMeasureCode, Name)
VALUES (src.UnitMeasureCode,
src.Name)
```

Inline Table-Valued Function

```
CREATE [ OR ALTER ] FUNCTION
<function_name> (<parameters>)
RETURNS TABLE
AS
RETURN
(
  <select_statement>
);
```

Multi-Statement Table-Valued Function

Check if table is existed (Using OBJECT_ID)

```
IF OBJECT_ID(N'dbo.
<table_name>', N'U') IS NOT NULL
BEGIN
    PRINT 1
END
```

add foreign key

```
CREATE TABLE <schema>.
<table_name>
   (
      <column_name> int NOT NULL
      , CONSTRAINT FK_<table_na-
me>_<reference_table_name>
FOREIGN KEY (<column_name>)
        REFERENCES <reference_t-
able_name> (<reference_column-
_name>)
        [ON DELETE CASCADE]
        [ON UPDATE CASCADE]
   )
;
```

add primary key

```
CREATE TABLE <schema>.
<table_name> (
      <column_name> int IDENTITY
(1,1) NOT NULL
      , CONSTRAINT PK_<table_na-
me>_<column_name> PRIMARY KEY
CLUSTERED (<column_name>)
   )
```

create unique constrant

```
CREATE TABLE <table_name>
(
   <column_name> int NOT NULL,
   CONSTRAINT UC_<column_name>
UNIQUE(<column_name>)
)
ALTER TABLE table_name
ADD CONSTRAINT constraint_name
UNIQUE (column1, column2, ...
column_n);
```

create index

```
CREATE [ OR ALTER ] FUNCTION
<function_name> ([ parameter ])
RETURNS @return_variable TABLE
<table_type_definition>
BEGIN
  <function_body>
  RETURN
END
```

```
CREATE TABLE <schema_name>.
<table_name>
(
    INDEX IX_<table_name>_<colu-
mn_name> NONCLUSTERED (<column_-
name>)
)
```

By **giangpdh**
cheatography.com/giangpdh/

Published 3rd August, 2021.
Last updated 17th June, 2022.