

2.T - SQL

Dátové typy a funkcie

- 1) Prvky T-SQL
 - 2) Dátové typy, pretypovanie
 - 3) Funkcie pre čísla, reťazce a dátumy
 - 4) CASE výraz
- #### Príklady

1) Prvky T-SQL

Jazyk T-SQL sa skladá z dvoch celkov. Prvý celok tvoria príkazy DDL, DML a SELECT, ktoré ilustrujú kódové riadky

```
USE tempdb
GO
```

```
DROP TABLE jajTab
CREATE TABLE jajTab(jaj int, [ha ha] int) -- [ha ha] <=> "ha ha"
GO
```

```
INSERT jajTab([ha ha], jaj) VALUES (1,11) -- 11 1
INSERT jajTab(jaj, [ha ha]) VALUES (2,22) -- 1 22
INSERT jajTab(jaj) VALUES (33) -- 33 NULL
INSERT jajTab("ha ha") VALUES (4) -- NULL 4
```

```
SELECT * FROM jajTab
```

Druhý celok T-SQL tvoria bežné prvky programovacích jazykov:

- identifikátory – stĺpcov, názvy Serverov, DB, Tab, pohľadov
 - “”, [], 128 znakov, #, @
- typy dát (dátových objektov) - stĺpcov, premenných, parametrov
- funkcie - IN, OUT - scalar/Tab; SUM, DATADIFF
- výrazy - skalárne: konštanty, sk. funkcie, odkaz na stĺpec, premenné
- operátory - pre výrazy; logické, porovnávacie, ...
- komentáre
- kľúčové slová

Nižšie sa podrobnejšie venujeme dátovým typom a niektorým štandardným funkciám T-SQL, ktoré ilustrujeme aj príkladmi.

2) Dátové typy, pretypovanie

Dátové typy aj v jazyku SQL zohrávajú dôležitú úlohu. Nastavenie nesprávneho typu stĺpca môže mať za následok pomalšiu odozvu dopytov, ale predovšetkým neefektívne využitie hardwarových kapacít. V SQL Server dátový typ majú aj premenné a parametre procedúr a funkcií.

Dátové typy rozdeľujeme na:

- Jednoduché - jednoduchá, skalárna hodnota – číslo, reťazec, ...
- Komplexné - sú skôr OO ako relačné.
- Špecializované - XML

Jednoduché dátové typy v SQL Server 2005:

- Presné numerické
 - BIT
 - TINYINT, SMALLINT, INT, BIGINT
 - DECIMAL (\Leftrightarrow NUMERIC)
 - MONEY, SMALLMONEY.
- Približné numerické
 - FLOAT, REAL
- Dátum a čas
 - DATETIME, SMALLDATETIME.
- Reťazce
 - CHAR, VARCHAR, TEXT.
- Unicode reťazce
 - NCHAR, NVARCHAR, NTEXT
 - BINARY, VARBINARY a IMAGE - Binárne reťazce tu nie sú dvojkové ale **hexadecimálne- šesťnásťkové (základ 16)**

Iné typy

- XML, CURSOR, TABLE, SQL_VARIANT, UNIQUEIDENTIFIER, TIMESTAMP.

Na pretypovanie slúžia príkazy CAST a CONVERT.

CAST(@x AS int) \Leftrightarrow CONVERT(int, @x)

```
SELECT CONVERT( BINARY(1), 255 ) --  $\Leftrightarrow$  :
SELECT CAST( 255 AS BINARY(1) ) -- 0xFF
SELECT CAST( 255 AS BINARY(2) ) -- 0x00FF
SELECT CAST( 255 AS BINARY(3) ) -- 0x0000FF
```

```
SELECT CAST( 12345678 AS BINARY(3) );
SELECT CAST( CAST( 12345678 AS BINARY(3) ) AS int ); -- OK - tam a späť
SELECT CAST( CAST( 123456789 AS BINARY(3) ) AS int ); -- NONO - tam a späť
```

Podrobne:

- Presné numerické
 - BIT
 - 1 bit: 0, 1
 - TINYINT, SMALLINT, INT, BIGINT
 - 1, 2, 4, 8 bytov; max: $255=2^8-1$, $2^{15}-1$, $2^{147\ 483\ 647}=2^{31}-1$, $2^{63}-1$ **Prečo nie napr. 2^{16} ??**
 - DECIMAL(c, d), NUMERIC
 - $1 \leq c \leq 38$, $0 \leq d \leq c$ - cifier
 - 28 platných cifier; 5-17 bytov; max: $10^{38}-1$
 - MONEY, SMALLMONEY
 - 8, 4 bytov; max: 922 337 203 685 477.5807
- Približné numerické (s pohyblivou desatinnou bodkou)
 - REAL, FLOAT(n) - default hodnota pre n je 53 bitov
 - REAL \Leftrightarrow FLOAT(24), FLOAT(53) – jednoduchá a dvojitá presnosť s 7 a 15 platnými ciframi; max: $3.40E+38$, $1.79E+308$
 - Ak $1 \leq n \leq 24$, n je 24. Ak $25 \leq n \leq 53$, n je 53.

Rozdiel medzi *približnými* a *presnými* číslami:

REAL, FLOAT a DECIMAL – 7, 15 a 28 platných cifier **Prečo nie 8, 16 ??**

```

DECLARE @d as DEC( 5, 2) -- DEC(4,2)
DECLARE @d2 as DEC(28,10) -- 28 cifier
SET @d = 123.456; SELECT @d -- 123.46x
SET @d2 = 123456789012345678.0123456789; SELECT @d2

DECLARE @r as REAL -- 7 cifier ↔ FLOAT(24)
SET @r = 123.456789E-01; SELECT @r -- 12.34568
SET @r = 1.0/3; SELECT @r -- 0.333333

DECLARE @f as FLOAT(53) -- 16 cifier
SET @f = 1123456789.0123456789E+08; SELECT @f -- 1.12345678901235E+17
SET @f = CAST(1 AS FLOAT(53))/3; SELECT @f -- 0.3333333333333333

```

- Dátum a čas

- DATETIME, SMALLDATETIME.

```

SELECT CAST('20101224' as DATETIME) -- 2010-12-24 00:00:00.000
-- ISO dateformat: yyyyymmdd je bezpečný dateformat, tzn. funguje nezávisle od nastavení PC.
Neovplyvnuju to ani SET DATEFORMAT ci SET LANGUAGE.
SELECT CAST(10 AS BINARY(1)) -- 0x0A
SELECT CAST(15 AS BINARY(2))
SELECT CAST(16 AS BINARY(8)) -- 0x0000000000000010
DECLARE @bin BINARY(8)
SET @bin = CAST(GETDATE() AS BINARY(8)) -----
SELECT @bin, CAST(@bin AS DATETIME) -- 0x00009DFF008AA704 2010-09-28 08:24:49.080

```

- Reťazce s pevnou a premenlivou dĺžkou

- char(n) , varchar(n) , varchar(max) - ASCII
 - nchar(k) , nvarchar(k) , nvarchar(max) - UNICODE - national char
- 1 ASCII znak je dĺžky 1 byte – 256 možností
1 UNICODE znak je 2 byte – 65536

1<= n <= 8000 => max: 1byte x 8000 = 8000 bytov = 8000 non-Unicode znakov

1<= k <= 4000 => max: 2byte x 4000 = 8000 bytov = 4000 Unicode znakov

- Default hodnote pre n je 1/30 v prípade deklarácie/convert,cast

- varchar(max) až 2GB, ako BLOBS

- Binárne reťazce

- BINARY, VARBINARY a IMAGE.

- Iné typy

- XML, CURSOR, TABLE, SQL_VARIANT, UNIQUEIDENTIFIER, TIMESTAMP.

- užívateľské

3) Funkcie pre čísla, reťazce a dátumy

[http://msdn.microsoft.com/en-us/library/ms174318\(SQL.90\).aspx](http://msdn.microsoft.com/en-us/library/ms174318(SQL.90).aspx)

Čísla - 23 funkcií:

ABS	CEILING	FLOOR	RADIANS	SQRT
ACOS	COS	LOG	RAND	SQUARE
ASIN	COT	LOG10	ROUND	TAN
ATAN	DEGREES	PI	SIGN	
ATN2	EXP	POWER	SIN	

```

SELECT ROUND(123.45, 1) -- 123.50
SELECT ROUND(123.45, -2) -- 100.00
SELECT ROUND(123.45, -3) -- 0.00 - cele

```

5 operácií

- + (Add): Addition
- - (Subtract): Subtraction
- * (Multiply): Multiplication
- / (Divide): Division
- % (Modulo): Modulation - `SELECT 17 % 5 -- 2`

Ret'azce

ASCII	LEN	REPLACE	STR	LOWER	REVERSE
CHAR	SUBSTRING	STUFF	DIFFERENCE	UPPER	SOUNDEX
UNICODE	LEFT	CHARINDEX	REPLICATE	LTRIM	QUOTENAME
NCHAR	RIGHT	PATINDEX	SPACE	RTRIM	

LEN vs. DATALENGTH – vrátia počet bytov pre reprezentácie výrazu

```
SELECT CHAR(64+1)      -- A
SELECT CHAR(64+32+1)  -- a
SELECT ASCII('A')     -- 65
SELECT ASCII(1)       -- 49
```

Špeciálne znaky:

```
PRINT 'a' + CHAR(9) + 'b' -- tab
PRINT 'a' + CHAR(10) + 'b' -- new line
PRINT 'a' + CHAR(13) + 'b' -- carriage return
-- vysledok:
a      b
a
b
a
b
```

- 1 ASCII (**American Standard Code for Information Interchange**) znak = 1 byte
- 1 A4 stránka obsahuje 46 riadkov po 72 znakov, čo predsta. 3312 ASCII znakov, 3.312 kB.
 - Notepad - 3.32 kB
 - MSWord 2003 – 28 kB
- 100 strán MSWord 2003 – 612 kB

Syntax: **CHARINDEX** (coho, vcom [, start.pozicia]) - PATINDEX
SUBSTRING (zcoho, start.pozicia, kolko) - LEFT, RIGHT
REPLACE (vcom, co, scim)
REPLICATE (co, kolkokrat)

```
SELECT SUBSTRING('Kosice', 2, 3)      -- osi
SELECT REPLACE('Bratislava', 'ratis', '') -- Blava
```

Pad - doplní koncové/okrajové

- medzere pre char stĺpce
- nuly pre binárne stĺpce

Trim - odstráni - opačný proces.

```

SET ANSI_PADDING OFF      -- default on
- CHAR stĺpce sa dopĺňajú s medzermi do požadovanej dĺžky
- VARCHAR stĺpce koncové medzery neodstránia (nie TRIM)
DECLARE @a AS VARCHAR(5)
DECLARE @b AS CHAR(5)
SET @a = 'ano'
SET @b = 'ano'

SELECT @a + '!'           -- ano!
SELECT @b + '!'           -- ano !
IF @a = @b SELECT '=' ELSE SELECT '<>' -- =

DECLARE @tr varchar(7) = '  ab'
SELECT LTRIM(@tr);        --ab

```

Dátumy

GETDATE	DAY	MONTH	YEAR	DATENAME *
DATEPART *	DATEADD *	DATEDIFF *	GETUTCDATE	

DATEPART * Skratka

year	yy, yyyy	hour	hh
quarter	qq, q	minute	mi, n
month	mm, m	second	ss, s
dayofyear	dy, y	Millisecond	ms
day	dd, d		
week	wk, ww		
weekday	dw		

Syntax: **DATEADD** (co*, kolko, datum)
DATEDIFF (co*, datum1, datum2)

```

SELECT GETDATE ()           -- 2008-09-21 21:49:19.090
SELECT DATEPART(month, GETDATE ()) -- 9
SELECT DATENAME (month, GETDATE ()) -- september

```

```

SET DATEFORMAT dmy;
GO
DECLARE @dt DATETIME;
SET @dt = '31.12.1998';
SELECT @dt                  -- 1998-12-31 00:00:00.000

```

```

SET DATEFIRST 7;           -- default
SELECT @@DATEFIRST AS 'Haha' -- 7
SELECT DATEPART(dw, GETDATE ()) AS 'Today' -- 1 (nedela)
SET DATEFIRST 1;
SELECT DATEPART(dw, GETDATE ()) AS 'Today' -- 7 (nedela)

```

```

DECLARE @Today AS DATETIME
SET LANGUAGE Slovak
SET @Today = '16.3.2008';
SELECT DATENAME(month, @Today) -- marec
SELECT DATENAME(weekday, @Today) -- nedela

```

```

SET LANGUAGE English
SELECT DATEDIFF(day, '2008.9.9', '2008.12.7') -- 89
SELECT DATEADD(day, 89, '2008.9.9') -- 2008-12-07 00:00:00.000

```

Typ výsledku operácie s dvomi rôznymi typmi implicitne je typ s vyššou prioritou. Ak nie je možné pretypovanie na vyšší typ, vráti sa chyba.

Collation určí sadu znakov a ich triedenie – **symbol ordering** / alphabetisation

3, -2, 2 => -2, 2, 3

a, A, \$ => \$, A, a

švédčina: z < ö

nemčina: ö < z

slovenčina: ch sú dva znaky

4) CASE výraz

Jednoduchý a vyhľadávaný CASE výraz:

a) CASE výraz WHEN c1 THEN ... - jednoduchý CASE výraz

b) CASE WHEN výraz=c1 THEN ... - vyhľadávaný CASE výraz

- Dovoľuje iba rovnosti.

- Môže vrátiť NULL.

CASE výraz je možné použiť

- v príkazoch SELECT, UPDATE, DELETE, SET a

- v klauzulách select_list, IN, WHERE, ORDER BY a HAVING.

---- Vráťte datum_narodenia, rok a jeho párnosť.

---- Return datum_narodenia, year and whether is it even or odd.

USE UPJS

```
SELECT datum_narodenia, YEAR(datum_narodenia)
    , CASE
        WHEN YEAR(datum_narodenia)%2 = 0 THEN 'Parny'
        WHEN YEAR(datum_narodenia)%2 = 1 THEN 'Neparny'
        ELSE '?'
    END as "Parnost"
FROM student
```

---- Vráťte datum_narodenia, rok a kvartál.

---- Return datum_narodenia, year and quartal.

USE UPJS

```
SELECT datum_narodenia, YEAR(datum_narodenia)
    , CASE
        WHEN Month(datum_narodenia) in(1,2,3) THEN 1
        WHEN Month(datum_narodenia) in(4,5,6) THEN 2
        WHEN Month(datum_narodenia) in(7,8,9) THEN 3
        WHEN Month(datum_narodenia) in(10,11,12) THEN 4
        --ELSE datum_narodenia
    END as "Kvartal"
FROM student
```

Implicitné pretypovanie

<http://technet.microsoft.com/en-us/library/ms187928.aspx>

<http://msdn.microsoft.com/en-us/library/ms187928.aspx>

From:	binary	varbinary	char	varchar	nchar	nvarchar	datetime	smalldatetime	date	time	datetimeoffset	datetime2	decimal	numeric	float	real	bigint	int(INT4)	smallint(INT2)	tinyint(INT1)	money	smallmoney	bit	timestamp	uniqueidentifier	image	ntext	text	sql_variant	xml	CLR UDT	hierarchyid
binary		●	●	●	●	●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	
varbinary	●		●	●	●	●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	
char	●	●		●	●	●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	
varchar	●	●	●		●	●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	
nchar	●	●	●	●		●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	
nvarchar	●	●	●	●	●		●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	
datetime	●	●	●	●	●	●		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	
smalldatetime	●	●	●	●	●	●	●		●	●	●	●	●	●	○	○	●	●	●	●	●	●	●	●	○	○	●	●	●	●	●	
date	●	●	●	●	●	●	●	●		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
time	●	●	●	●	●	●	●	○	○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
datetimeoffset	●	●	●	●	●	●	●	○	○	○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
datetime2	●	●	●	●	●	●	●	○	○	○	○		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
decimal	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
numeric	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
float	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
real	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
bigint	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
int(INT4)	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
smallint(INT2)	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
tinyint(INT1)	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
money	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
smallmoney	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
bit	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
timestamp	●	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
uniqueidentifier	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
image	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
ntext	○	○	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
text	○	○	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
sql_variant	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	○	○	○	○	○	○	○	
xml	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
CLR UDT	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	
hierarchyid	●	●	●	●	●	●	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	

- Explicit conversion
- Implicit conversion
- Conversion not allowed
- ✱ Requires explicit CAST to prevent the loss of precision or scale that might occur in an implicit conversion.
- Implicit conversions between xml data types are supported only if the source or target is untyped xml. Otherwise, the conversion must be explicit.

Odkaz na dátový zdroj

Úplné meno objektu sa skladá zo štyroch častí:

- server.database.schema.object
- server...object
- database..object

`SELECT * FROM dbo.jajTab` - *dbo* je skratka z *database owner*