# IT 775
# Database Technology

## SQL

## Data Types

# Numeric Data Types

- integer

| Type | Storage (Bytes) | Minimum Value (Signed/Unsigned) | Maximum Value Signed/Unsigned) |
|---|---|---|---|
| TINYINT | 1 | -128 | 127 |
| | | 0 | 255 |
| SMALLINT | 2 | -32768 | 32767 |
| | | 0 | 65535 |
| MEDIUMINT | 3 | -8388608 | 8388607 |
| | | 0 | 16777215 |
| INT | 4 | -2147483648 | 2147483647 |
| | | 0 | 4294967295 |
| BIGINT | 8 | -9223372036854775808 | 9223372036854775807 |
| | | 0 | 18446744073709551615 |

# Numeric Data Types

- decimal
  - decimal[(p [, s])] p digits precision, s of them fractional (scale)
  - numeric[(p [, s])] p digits precision, s of them fractional (scale)
    - » same as decimal
- real (approximate values)
  - float 24, single precision
  - double 53, double precision

  - NON-STANDARD:
  - float(p, s) p digits precision, s of them fractional (scale)
  - double(p, s) p digits precision, s of them fractional (scale)

- bit(m) m digits binary

# String Data Types

- fixed length,
  - char[acter][( n )] 1 – 8000 chars, 1 is the default
    n bytes allocated
- variable length
  - varchar[acter][( n )] 1 – 8000 chars, 1 is the default
    bytes allocated to hold actual string

| Value | CHAR(4) | Storage Required | VARCHAR (4) | Storage Required |
|---|---|---|---|---|
| '' | '  ' | 4 bytes | '' | 1 byte |
| 'ab' | 'ab  ' | 4 bytes | 'ab' | 3 bytes |
| 'abcd' | 'abcd' | 4 bytes | 'abcd' | 5 bytes |
| 'abcdefgh' | 'abcd' | 4 bytes | 'abcd' | 5 bytes |

# Temporal Data Types

- date — date only, no time of day
  3 bytes: 0001-01-01 – 9999-12-31
  constants ' YYYY-MM-DD'
- time — time only, no date
  '-838:59:59' to '838:59:59‘ shows time of day or elapsed time
  constants ' HH:MM:SS' ‘HHH:MM:SS'
- datetime — date and time 6 – 8 bytes
  '1000-01-01 00:00:00' to '9999-12-31 23:59:59‘
  constants YYYY-MM-DD HH:MM:SS'
  ANSI standard name is TIMESTAMP

# Date Comparison

date types compare with usual comparison operators

all forms of literals are equivalent

for date comparisons of different types inferior type cast to superior type

Examples:

'10-01-2010' < '10-02-2010'

'10-01-2010' < '10-01-2011'

'01-oct-2010' = '10-01-2010'

'10-01-2010' < '10-01-2010 01:01:01'

- date promoted to datetime for comparison

# Data Type Conversion

## implicit conversion

- SQL hides from end user
  - assigning value to column –
    - converts value to column type
  - expression with differently typed arguments
    - result has higher type
- implicit conversions
  - from lower precedence type to higher
  - within type
    - each precision/length is a different type
    - more precision/length is higher

## explicit conversion

- when implicit conversion isn't available
- CAST and CONVERT do explicit conversion

highest
---
datetime
(timestamp)
smalldatetime
date time
float
real
decimal
int
smallint
tinyint
bit
varchar
char
---
lowest

# CAST, CONVERT and String Functions

cast is SQL standard type

CAST( expr AS type )

convert is unique to MySQL, not portable – avoid

LTRIM( string ), RTRIM( string )

returns string with leading/trailing spaces removed
ANSI SQL specifies TRIM which does both (in MySQL)

SUBSTR( string, start, length )

returns portion from start position for length characters

LOWER( string ), UPPER( string )

modifies case of characters in the string

# Date and Time Functions

GETDATE( ) returns today's date

SYSDATETIME( ) reurns current local date and time

SYSDATETIMEOFFSET( ) returns date, time, & offset

DAY( date ) returns day of month as int

MONTH( date ) returns month as int

YEAR( date ) returns year as 4-digit int

DATEPART( datepart, date )

DATEPART( month, '2010-10-01' ) => 10

DATENAME( month, '2010-10-01' ) => october

day, month, year, hour, minute, second, quarter, dayofyear, week, weekday, millisecond, microsecond, nanosecond, tzoffset

DATEADD( datepart, number, date )

DATEADD( hour, 2, '2013-10-01 13:05:44' ) => '2013-10-01 15:05:44'

DATEDIFF( datepart, startdate, enddate )

DATEDIFF( hour, 2, '2013-10-01 13:05:44', '2013-10-01 15:05:44' ) => 2

searching for date values

SELECT … WHERE MONTH( date ) = 9 AND YEAR( date ) = 2013