5.2: Data types

IT2306 – Database Systems I

Level I - Semester 2

Detailed Syllabus

- 5.2.1 Categories of Data Types:
  - Character, Numeric, Variable Character, Date, Serial, Money, Date-time, Interval.
- 5.2.2
  - Character: CHARACTER (CHAR);
  - Numeric: INTEGER (INT), SMALLINT, FLOAT, SMALLFLOAT, DECIMAL;
  - Variable Character: CHARACTER VARYING (VARCHAR);
  - Binary Large Object (BLOB): Text, Byte.

Attribute Data Types and Domains in SQL

- The basic data types available for attributes:
  - numeric,
  - character string,
  - bit string,
  - Boolean,
  - date,
  - and time

Attribute Data Types and Domains in SQL

- **Numeric data types:**
  - integer numbers of various sizes
    (INTEGER or INT, and SMALLINT)
  - floating-point (real) numbers of various precision
    (FLOAT or REAL, and DOUBLE PRECISION).
  - Formatted numbers can be declared by using DECIMAL(*i, j*)—or DEC(*i, j*) or NUMERIC(*i, j*)—where *i*, the *precision*, is the total number of decimal digits and *j*, the *scale*, is the number of digits after the decimal point.
  - The default for *scale* is zero, and the default for *precision* is implementation-defined.

Attribute Data Types and Domains in SQL

- **Character-string data types**
  - Fixed length
    - CHAR(n) or CHARACTER(n), where n is the number of characters
    - a shorter string is padded with blank characters to the right.
    - For example, if the value ‘Smith’ is for an attribute of type CHAR(10), it is padded with five blank characters to become ‘Smith’ if needed.
    - Padded blanks are generally ignored when strings are compared.
  - Varying length
    - VARCHAR(n) or CHAR VARYING(n) or CHARACTER VARYING(n), where n is the maximum number of characters.

Attribute Data Types and Domains in SQL

- **Character-string data types**
  - When specifying a literal string value, it is placed between single quotation marks (apostrophes), and it is case sensitive.
  - Strings are considered ordered in alphabetic (or lexicographic) order;
    - if a string str1 appears before another string str2 in alphabetic order, then str1 is considered to be less than str2.
  - There is also a concatenation operator denoted by || that can concatenate two strings in SQL.
    - For example, ‘abc’ || ‘XYZ’ results in a single string ‘abcXYZ’.
  - Another variable-length string data type called CHARACTER LARGE OBJECT or CLOB is also available to specify columns that have large text values, such as documents.
    - The CLOB maximum length can be specified in kilobytes (K), megabytes (M), or gigabytes (G).
    - For example, CLOB(20M) specifies a maximum length of 20 megabytes.

Attribute Data Types and Domains in SQL

- **Bit-string data types**
  - Fixed length n - BIT(n)
  - Varying length - BIT VARYING(n), where n is the maximum number of bits.

  - The default for n (the length of a character string or bit string) is 1.
  - Literal bit strings are placed between single quotes but preceded by a B to distinguish them from character strings;
    - for example, B‘10101’.

Attribute Data Types and Domains in SQL

- **Bit-string data types**
  - BINARY LARGE OBJECT (BLOB)
    - Another variable-length bit-string data type
    - Available to specify columns that have large binary values, such as images.
    - As for CLOB, the maximum length of a BLOB can be specified in kilobits (K), megabits (M), or gigabits (G).
    - For example, BLOB(30G) specifies a maximum length of 30 gigabits.

Attribute Data Types and Domains in SQL

- **Boolean data type**
  - Has the traditional values of TRUE or FALSE.
  - In SQL, because of the presence of NULL values, a three-valued logic is used,
  - Third possible value for a Boolean data type is UNKNOWN.

Attribute Data Types and Domains in SQL

- **DATE data type**
  - Has ten positions,
  - Its components are YEAR, MONTH, and DAY in the form YYYY-MM-DD.
- **TIME data type**
  - Has at least eight positions,
  - The components HOUR, MINUTE, and SECOND in the form HH:MM:SS.

Attribute Data Types and Domains in SQL

- **Date & Time**
  - Only valid dates and times should be allowed by the SQL implementation.
  - Months should be between 1 and 12
  - days must be between 01 and 31;
  - A day should be a valid day for the corresponding month.
  - The < (less than) comparison can be used with dates or times - an earlier date is considered to be smaller than a later date, and similarly with time.

Attribute Data Types and Domains in SQL

- **Date & Time**
  - Literal values are represented by single-quoted strings preceded by the keyword DATE or TIME;
    - for example, DATE ‘2014-09-27’ or TIME ‘09:12:47’.
  - A data type TIME(*i*), where *i* is called *time fractional seconds precision*, specifies i + 1 additional positions for TIME
    - one position for an additional period (.) separator character,
    - and i positions for specifying decimal fractions of a second.

Attribute Data Types and Domains in SQL

- **Date & Time**
  - **TIME WITH TIME ZONE data type**
    - **I**ncludes an additional six positions for specifying the displacement from the standard universal time zone, which is in the range +13:00 to -12:59 in units of HOURS:MINUTES.
    - If WITH TIME ZONE is not included, the default is the local time zone for the SQL session.

Attribute Data Types and Domains in SQL

- **Timestamp data type** (TIMESTAMP)
  - Includes the DATE and TIME fields,
  - A minimum of six positions for decimal fractions of seconds
  - An optional WITH TIME ZONE qualifier.
  - Literal values are represented by single-quoted strings preceded by the keyword TIMESTAMP, with a blank space between data and time; for example, TIMESTAMP ‘2014-09-27 09:12:47.648302’.

Attribute Data Types and Domains in SQL

- **INTERVAL data type**
  - This specifies an interval—a relative value that can be used to increment or decrement an absolute value of a date, time, or timestamp.
  - Intervals are qualified to be either YEAR/MONTH intervals or DAY/TIME intervals.

Attribute Data Types and Domains in SQL

- The format of DATE, TIME, and TIMESTAMP can be considered as a special type of string.
- Hence, they can generally be used in string comparisons by being cast (or coerced or converted) into the equivalent strings.
- It is possible to specify the data type of each attribute directly
- Alternatively, a domain can be declared, and the domain name can be used with the attribute specification.
- This makes it easier to change the data type for a domain that is used by numerous attributes in a schema, and improves schema readability