Field Types in Tables

Database tables in TNTgis can include fields with a variety of data types. The data type for each field in a table is shown in the Type column in the Table Properties window. When you add fields to a table you choose the data type from the menu provided by the Add Field icon button (see the Technical Guide entitled *Table Properties Window*). This Technical Guide describes the properties and limits of the various field data types.

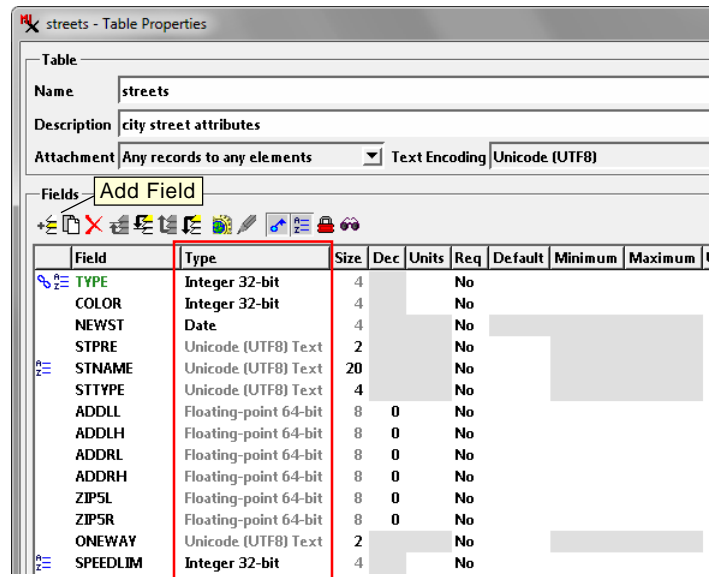| Numeric Field Types		
Data Type	**Numeric Range**	**Other Common Names**
Integer 8-bit	-128 to 127	TINYINT
Integer 16-bit	-32,768 to 32,767	SMALLINT
Integer 32-bit	-2,147,483,648 to 2,147,483,647	INTEGER, INT
Integer 64-bit	-2^{63} to $2^{63} - 1$	BIGINT
Unsigned 8-bit	0 to 255	TINYINT
Unsigned 16-bit	0 to 65,535	SMALLINT
Floating-point 32-bit (6-digit precision)	±1e-37 to 1e+37	REAL, FLOAT
Floating-point 64-bit (15-digit precision)	±1e-307 to 1e+308	DOUBLE

Numeric Field Types

Several types of numeric field are provided for storing integer or decimal values of different ranges, as detailed in the table above. The larger the bit-depth of the field, the larger the range of numbers that it can hold, but this extended range comes at the expense of a larger storage space requirement for each value. To minimize the stored size of the database, choose the smallest numeric data type needed for the anticipated range of stored values. The Size of a numeric field is automatically set large enough to accommodate the maximum number of digits needed.

Integer: Integer fields store positive or negative whole numbers (without decimal fractions). You can choose from four Integer field types depending on the range of the values you need to store. Enter a minus sign before the integer value when entering a negative number. Plus signs are not shown for positive values and need not be entered with the value. The default value of an integer field is 0. Decimal values entered manually into or written to an integer field are truncated to the integer value.
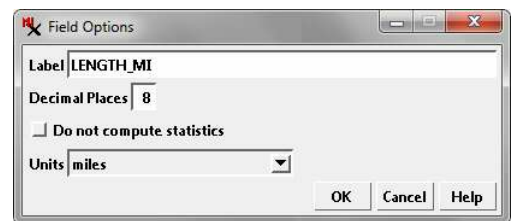
Unsigned: The unsigned field types are limited to positive integer values. The default value for an unsigned integer field is 0.

Floating-Point: Floating-point fields store decimal (fractional) values with variable precision. The precision value shown for the two floating-point field types indicates the number of decimal places allowed to the right of the decimal point. The default value of a floating-point field is 0.

Use a floating-point field for any decimal numeric value. Numbers can be entered in floating-point fields using standard notation or in the computer equivalent of scientific notation. For example, the value 7847.365 could also be entered as 7.847635e+003.

The number of decimal places to display in table views can be set in two ways. You can enter the desired number of places in the *Dec* column in the Table Properties window or in the Decimal Places field in the Field Options window (opened from the table view by right-clicking on the field name and choosing Options from the menu).

Other Database Field Types		
[encoding] Text	Auto-increment	Date
Unicode Text	Logical	Date/Time (Local)
Single Character	Binary	Date/Time (UTC)
Memo	Color	

Auto-Increment Fields

An auto-increment field is a 32-bit integer field that is automatically assigned a sequential integer value when a new record is added. These fields are used for automatically creating a unique identifier for each record in the table. When adding an auto-increment field to a table with existing records, use the Populate Database Records operation to create sequential values in the new auto-increment field for all of the records (see the TechGuide entitled *Populate Database Records* for more information).
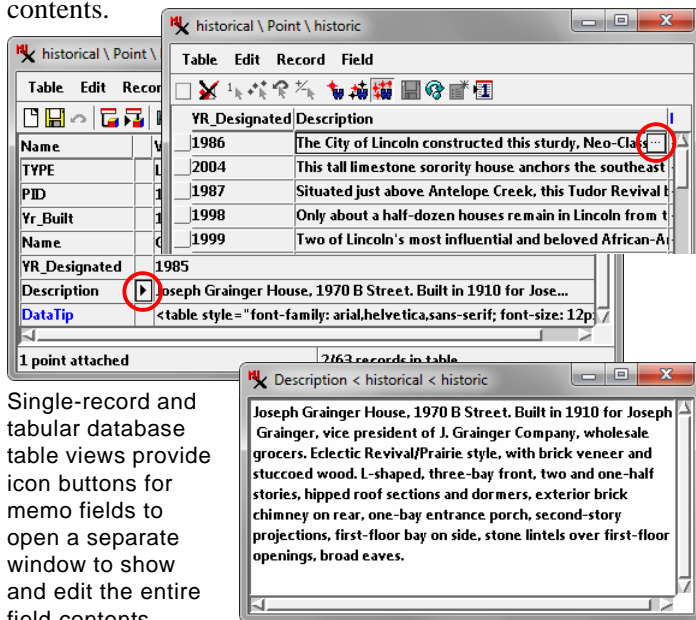
(over)

Text Field Types

The Text, Unicode Text, Memo, and Single Character field types store text strings. These fields can store any character you can type on your keyboard. Any number entered into one of these text fields is treated as text rather than as a number.

Text: The Text field type stores character strings using the character encoding specified for the table. In the Add Field menu this entry is preceded by the specified table encoding, *e.g.*, ASCII Text. The Text field type can store character strings of varying length up to the maximum set by the field Size setting.

Unicode Text: this field type stores characters strings using Unicode (2-byte) encoding. The number of characters in a stored string is limited by the field Size setting.

Memo: A Memo field stores text with no specified limit to the number of characters stored. Memo fields can be used for lengthy observations, descriptions, or other notes that expand on the information stored in the other fields and may not be readily categorized or reduced to fit in a smaller text field. In practice you can store a few paragraphs to several pages of text in a memo field. Tabular and single-record table views show the first part of the memo field contents by default and provide special controls for viewing and editing the entire contents. In tabular views a small icon button appears in the field when you left-click in it, and memo fields in single-record views are provided with a small arrow icon button to the right of the field. Clicking on these icon buttons opens a separate window for viewing, entering, or editing the entire memo field contents.
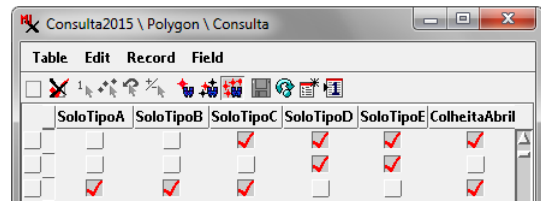

Single-record and tabular database table views provide icon buttons for memo fields to open a separate window to show and edit the entire field contents.

Logical Fields

A logical field is used to record a state of True or False, Yes or No. Tabular views show the field state with a checkbox that can be checked on (check mark shown) for True and checked off (check mark not shown) for False (see illustration in the next column). In single-record views the field shows either "Yes" or "No" (or their equivalents if the TNTgis interface is

set to use a different language). A Yes state can be entered as: yes, true, y, t, or 1; a False state can be entered as no, false, n, f, or 0. Capitalization is ignored, and localized text strings (and their first characters) can be used. Any entered text that cannot be parsed results in a state of "undefined". Logical states are stored as binary numeric values (1 or 0), and queries using logical fields must treat the field as numeric.

Date and Date/Time Fields

Date fields store a calendar date. The Field Options window for Date fields provides a menu from which you can choose the format for showing and entering the date; the formats and examples are listed below.

Format	Example	Format	Example
YYYYMMDD	21050523	M-D-YYYY	5-23-2015
M/D/YYYY	5/23/2015	D-M-YYYY	23-5-2015
D/M/YYYY	23/5/2015	YYYY-M-D	2015-5-23
YYYY/M/D	2015/5/23	D-Mmm-YYYY	23-May-2015
		D-Mmm-YY	23-May-15

Date/Time fields store a date and a time, which are shown in a fixed format:

MM/DD/YY HH:MM:SS 05/14/15 16:40:29

A Date/Time (Local) field is provided for times that reference a single local time zone, while a Date/Time (UTC) field is provided for times relative to Universal Coordinated Time, sometimes referred to as Greenwich Mean Time (GMT). Entries in a Date/Time (Local) field do not include the time zone or offset to UTC. The time zone used can be listed in the object metadata if this information is important.

Binary Fields

Binary fields store binary image data in tables created by the Create Geotagged Image Database process (see the TechGuide entitled *Geotagging Digital Photos*).

Color Fields

A Color field stores a color specification in the form of red, green, and blue values (range 0 to 255 for each component). Color fields are provided with a color button showing the color; left-clicking on the color button opens a standard Choose Color window allowing the user to select a color from a palette or design a custom color. The value of a Color field can be read by an SML script and assigned to a COLOR class instance for use in processing the parent object.

Computed Fields

Numeric and text fields can be set to be *computed*, which means that the field does not contain static data but instead uses an expression to compute a value from values stored in one or more other fields in the same table or a related table. See the TechGuide entitled *Computed Numeric and Text Fields* for more information.