

数値処理フィールドと文字処理フィールド

①計算フィールドはデータベーステーブルのフィールドで、固定のデータを持たない代わりに、式を使って計算した値を表示します。同じテーブルやリレートしたテーブル内の他のフィールドにある値を使って計算します。計算フィールドの式から、テーブルの各レコードの値が別々に求められるので、同じ式でもレコードごとに異なる値を返すことができます。計算フィールドには2つのタイプがあります。数値処理と文字処理フィールドです。

②他のフィールドを参照するだけでなく、計算フィールドでは算術演算子や定数を使ったり、可変値の定義や使用、戻り値を定義する条件付き論理を使用することが出来ます。テーブルウィンドウの中で、計算フィールドの名前は青で表示されるので、通常のデータフィールドと簡単に区別できます。計算フィールドの値は、参照フィールドのどこかで値が変更されると自動的に更新されます。表示された式の結果は、そのフィールドに関係する操作にも自動で使われます。例えば、他の形式(テキスト、CAD、HTML)へのテーブルの保存や、他のデータベース形式へのテーブルのエクスポート、別の計算フィールドの式からその計算フィールドを参照する時などに使用されます。

③計算フィールドを使用すると、他のフィールドからのテキストをつなげてダイナミックラベルやデータチップとして利用したり、他のテーブルの値を使用して簡単または複雑な数値処理を実行することができます。また、新規テーブルにおいて、沢山のリレートしたテーブルの中から重要な(際立った)データを表示するために使うことが出来ます。

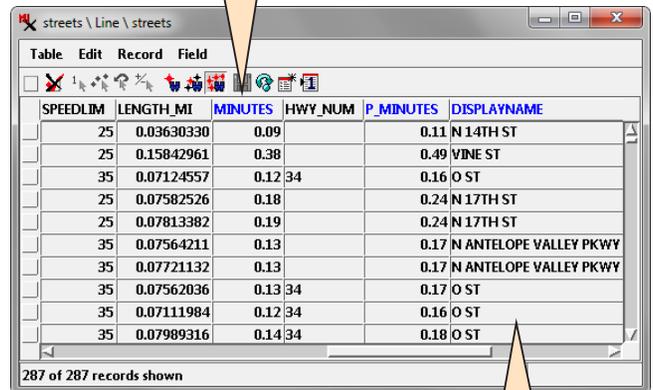
④計算フィールドの作成

数値処理と文字処理フィールドは、独立したフィールドタイプとして定義されている訳ではありません。さまざまなタイプの数値フィールドと文字フィールドの持つ特性として定義されています。例えば、数値処理フィールドにはいくつかの数値フィールドタイプがあり、フィールドの式から返る厳密な値はデータタイプによって異なります。同じ数値式でも、浮動小数点フィールドには小数点を持つ値が入り、整数フィールドには整数値が入ります。

⑤テーブルに計算フィールドを作成するには(右図参照):

- テーブルの〈テーブルプロパティ〉ウィンドウを開きます。
- [フィールドの追加 (Add Field)] アイコンボタン  を押して、希望の数字または文字データタイプを選択します。
- リストからフィールドを選択した状態で、[計算処理 (Computed)] アイコンボタン  を押します。
- 〈クエリ (Query Editor)〉 が自動で開き、計算処理フィールドに数値式や文字式を入力できます。
- 式を入れ終わったら [OK] を押して 〈クエリ〉 を閉じます。
- 計算処理機能を持つフィールドは、フィールド名が青で示されます。フィールド名の隣のステータス列には計算フィールドインジケータ  が表示されます。
- リストからフィールドを選んで [スクリプトの編集 (Edit Expression)] アイコンボタン  を押すことで、再び 〈クエリ〉 を開き、式を編集することができます。

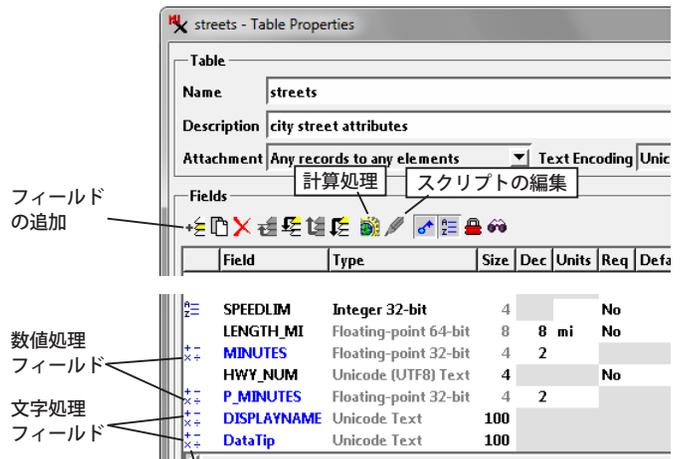
数値処理フィールド (Computed Numeric Field)
streets.LENGTH_MI * 60 / streets.SPEEDLIM;



SPEEDLIM	LENGTH_MI	MINUTES	HWY_NUM	P_MINUTES	DISPLAYNAME
25	0.03630330	0.09		0.11	N 14TH ST
25	0.15842961	0.38		0.49	VINE ST
35	0.07124557	0.12	34	0.16	O ST
25	0.07582526	0.18		0.24	N 17TH ST
25	0.07813382	0.19		0.24	N 17TH ST
35	0.07564211	0.13		0.17	N ANTELOPE VALLEY PKWY
35	0.07721132	0.13		0.17	N ANTELOPE VALLEY PKWY
35	0.07562036	0.13	34	0.17	O ST
35	0.07111984	0.12	34	0.16	O ST
35	0.07989316	0.14	34	0.18	O ST

文字処理フィールド (Computed Text Field)
string stname\$ = streets.STNAME + " " + streets.STTYPE;

if (streets.STPRE <> "")
return streets.STPRE + " " + stname\$;
else
return stname\$;



Field	Type	Size	Dec	Units	Req	Defa
SPEEDLIM	Integer 32-bit	4			No	
LENGTH_MI	Floating-point 64-bit	8	8	mi	No	
MINUTES	Floating-point 32-bit	4	2		No	
HWY_NUM	Unicode (UTF8) Text	4			No	
P_MINUTES	Floating-point 32-bit	4	2		No	
DISPLAYNAME	Unicode Text	100			No	
DataTip	Unicode Text	100			No	

フィールドの追加

数値処理フィールド

文字処理フィールド

計算処理フィールドステータスインジケータ

⑥簡単計算フィールド

計算処理フィールドの式が、他のフィールドを参照するだけの時のように単純な場合もあるでしょう。このような単純な計算フィールドを効率的に作るために、[フィールドの追加] アイコンボタンメニューには [簡単計算 (Simple Computed)] というオプションが備えられています。このオプションを使ってフィールドを追加すると、プロンプトが出てソースのテーブルとフィールドを選べます。ソースフィールドと同じ名前を持つ新規フィールドが作られます。参照フィールドに選んだフィールドのタイプに基づき、数値処理または文字処理フィールドとして作られます。数値と文字のデータタイプも、ソースフィールドから自動で設定されます。

⑦ [簡単計算] オプションで作成する計算処理フィールドは、独立したフィールドタイプではありません。デフォルトで参照フィールドの中身(値)がフィールドに表示されますが、もし希望があれば、〈クエリ〉を開いてさらに複雑な式を作成することができます。

⑧ 簡単計算オプションが特に役に立つのは、テーブル全体を計算フィールドで構成し、個々のリレートした他のテーブルから特に重要なデータを表示する場合です。このようなテーブルは、企業のデータベースの“テーブル表示”に似ています。データのプレゼンテーションを(参照元のテーブルは非表示にして)簡素化したり、また重要な属性情報をエクスポートしやすくするために作成することもあるでしょう。このようなテーブルを作成する時は、アタッチメントタイプを[各要素に対して厳密に1レコード(Exactly one record for every element)]にします。各要素に1つのレコードが自動で作られます(テクニカルガイド「レコードのアタッチメントタイプ」をご覧ください)。これらのレコードには、フィールドの式によって値を入れることができます。

⑨ 数値処理フィールド

数値処理フィールドは、どんな複雑な計算でも実行できます。下図は、2010年と2000年のネブラスカ州の郡の人口データのテーブルです。数値処理フィールドは、2010年の平方キロメートル当たりの人口密度と、10年間の人口の変化と割合の変化を示しています。

$NECntyPop2.POP_2010 / NECntyPop2.AREA_KM2;$

$NECntyPop2.POP_2010 - NECntyPop2.POP_2000;$

FIPS	COUNTY	AREA_KM2	POP_2010	POP_DENS	POP_2000	POP_CHNG	PCT_CHNG
31001	Adams	1514.7	31364	20.7	31151	213	0.7
31003	Antelope	2182.8	6685	3.1	7452	-767	-10.3
31005	Arthur	1860.4	460	0.2	444	16	3.6
31007	Banner	1946.3	690	0.4	819	-129	-15.8
31009	Blaine	1837.0	478	0.3	583	-105	-18.0
31011	Boone	1760.1	5505	3.1	6259	-754	-12.0
31013	Box Butte	2798.9	11308	4.0	12158	-850	-7.0
31015	Boyd	1442.9	2099	1.5	2438	-339	-13.9

$100 * NECntyPop2.POP_CHNG / NECntyPop2.POP_2000;$

⑩ TNTmipsの流水解析(Waterched)処理は、数値処理フィールドを使用して、いくつかの水理学的な属性を計算します。水路ライン・分水界ポリゴン・流域ポリゴンのテーブルの他のフィールドにある値から計算することができます。例えば、左

$2 * \sqrt{SetSum(POLYSTATS[*].Area) / PI} / ATTRIBUTES.'BASIN_LENGTH'$

BASIN_LENGTH	FORM_RATIO	ELONGATION_RATIO	BASIN_RELIEF	RELIEF_RATIO	RUGGEDNESS
1672.43	0.3234	0.64	867.00	0.52	
2959.65	0.0773	0.31	586.00	0.20	
2573.59	0.3844	0.70	1289.00	0.50	
1938.38	0.5461	0.83	1355.00	0.70	
1462.42	0.1662	0.46	753.00	0.51	

下の図の分水界ポリゴンの ATTRIBUTES テーブルは、1) 分水界の面積と同じ面積の円の直径の、2) 流域の長さ、水路の河口から測った最大のポリゴン面積、に対する比を計算した ELONGATION RATIO 計算処理フィールドを含みます。

⑪ 文字処理フィールド

文字処理フィールドを使って、さまざまなフィールドからのテキストを組み合わせて一つの文字列にすることが出来ます。例えば、所在地住所の構成要素(番号、道路の方向、道路名、道路タイプ)が別々のフィールドにある時、文字処理フィールドを使って、ダイナミックラベルやデータティップとして使う完全なアドレスを表示できます。さまざまなフィールドからの属性情報と共にテキストや改行コード("\n")を組み合わせて複数行のテキストを作成し、データティップに表示することが出来ます(下図)。テーブル表示では一行目のテキスト行しか表示されませんが、データティップとして表示した場合は全ての行が表示されます。

MAPUNITS.PTYPE + ", " + MAPUNITS.UNIT + "\n" +
"Origin: " + MAPUNITS.CAT2 + "\n" +
"Age: " + MAPUNITS.AGE;

文字処理フィールドの文字式のサンプル。他のフィールドの情報を組み合わせて、データティップ用の複数行のテキストにします。



⑫ HTML形式も、データティップとしての使用を目的とした文字式に適用することができます。詳しくは「HTMLを使ったデータティップ(Design DataTIPS using HTML)」と題されたテクニカルガイドをご覧ください。

⑬ 文字処理フィールドの文字式では、上に示されているように“+”や“+=”演算子を使って文字をつなげることができます。または、sprintf()やその他の文字処理関数を使ってフォーマットされたテキストを作成できます。数値フィールドを参照する式では、数字は必ず文字に変換する必要があります。

⑭ 計算フィールドのスクリプトは、条件付きテストを組み込むことが出来ます。式に使われるフィールドの内容に応じて、異なる結果を生成します。このようなスクリプトではreturnキーワードを使って希望の結果を明確に返す必要があります。前のページの図の、道路名の文字処理例では、いくつかの道路は道路の方向を示す接頭語を持っていますが、持っていないものもあります。DISPLAYNAMEフィールドの文字式では、初めに道路名とタイプを(間にスペースを置いて)組み合わせて、可変値sname\$を作っています。STPREフィールドが空でないレコードについては、接頭語と可変値が組み合わされて戻ります。それ以外の場合(空の場合)、可変値sname\$が戻ります。

⑯ 計算処理フィールドの固定化

一度計算フィールドを作成し、元にしたフィールドの値を変更する予定が無い場合、計算フィールドの結果を固定の不変値に変換するオプションがあります。テーブルをテーブル表示して、フィールド名(列の先頭)の上で右クリックし、メニューから[計算フィールドを固定化(Make field permanent)]を選択します。