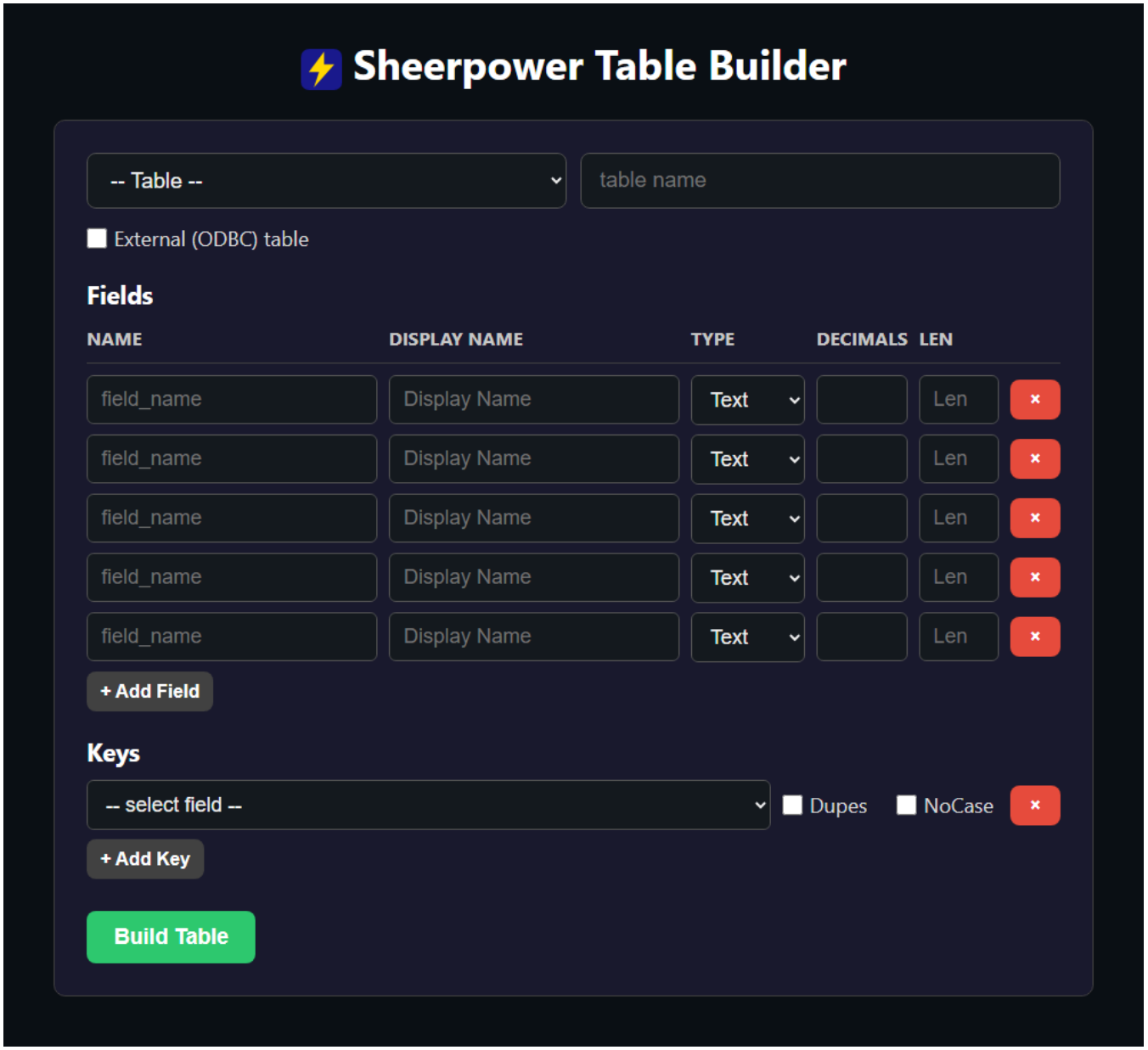
Sheerpower Table Builder

A visual, browser-based tool for defining database tables — no hand-editing of `.tdef` or `.def` files. It builds two kinds of tables:

- **Native Sheerpower (ARS) tables** — generates the full set of structure files (`.def`, `.fdl`, `.ARS`, `.str`) from a simple field/key form.
- **External (ODBC) table references** — generates a single `.str` manifest that points to a table living in *any* ODBC data source (SQL Server, PostgreSQL, Access, ...). The columns are read live from the driver.



Why use it

- **No structure-file syntax to learn.** Fill in field names, types, lengths, and keys; the tool writes the files and runs the build for you.
- **It validates as you go.** Required names, numeric decimals, and lengths are

checked in the form (with an audible cue on bad input), so you find mistakes before you build.

- **It learns your common fields.** Field names you use are remembered, so the next table auto-fills the display name, type, length, and decimals as you type.
- **Tables are immediately usable everywhere.** Anything you build drops into the folder and is instantly discoverable by other Sheerpower apps — for example it shows up in the Report Writer's table list with no extra steps.
- **Reach data that lives in other databases.** With an ODBC reference you can open, search, and report on a table from another system — SQL Server, PostgreSQL, Access, and more — just like one of your own. Amounts and prices come back exact, with no rounding surprises.

Getting there

Start the handler (it launches the web server if it isn't already running):

```
sp4gl /run "table_builder_web.spsrc"
```

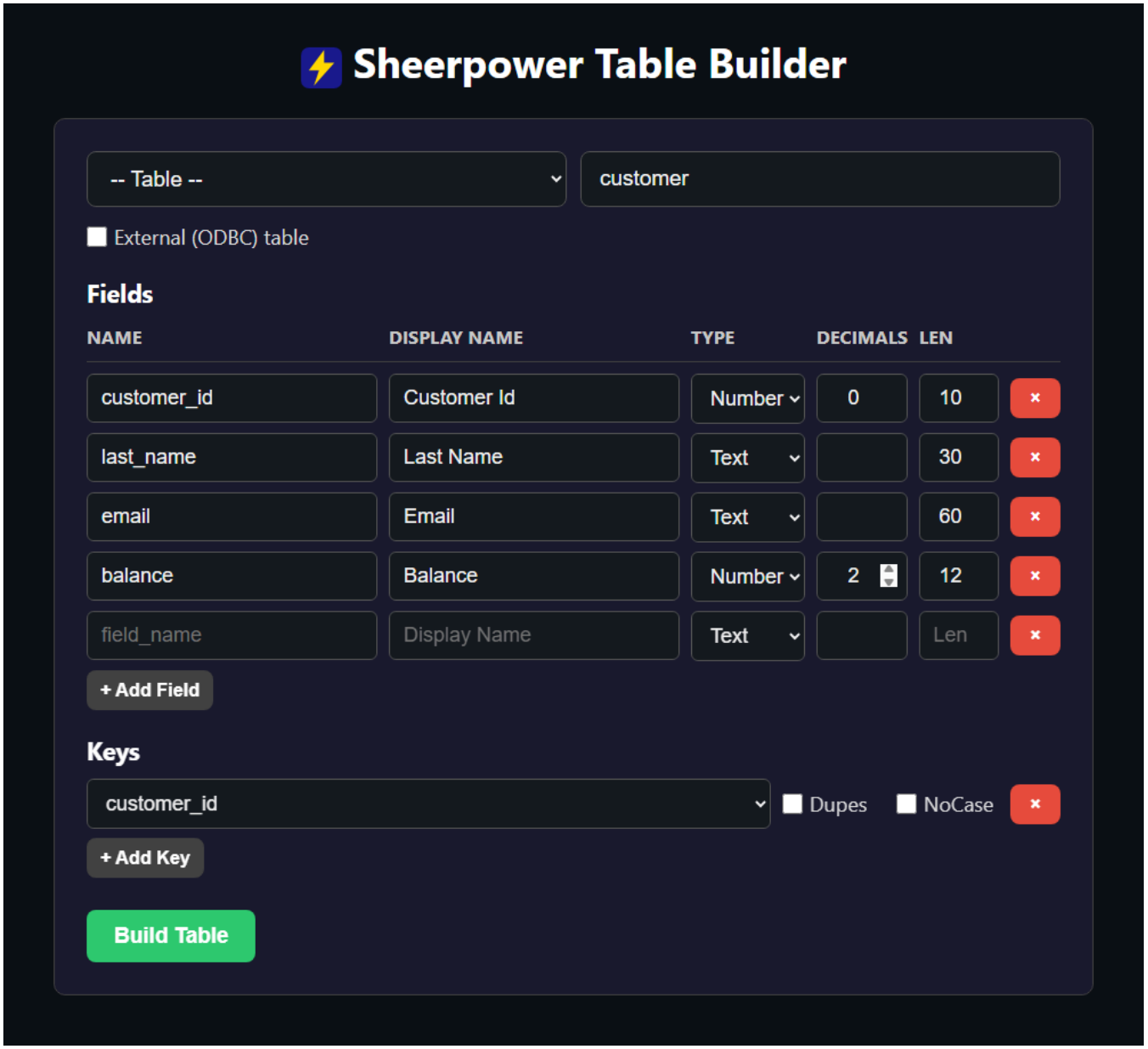
Then open it in a browser:

```
http://<your-host>/table_builder_web.html
```

No login is required, and the builder itself uses no database tables.

Building a native table

1. **Name the table** in the box at the top right (e.g. `customer`). Leave the **External (ODBC) table** checkbox unchecked.
2. **Add the fields.** For each field, give a **Name** (the display name auto-fills from it), a **Type** (Text or Number), a **Len** (storage length), and for numbers a **Decimals** count. Use **+ Add Field** for more rows and the red **×** to remove one.
3. **Choose the keys.** Pick a field for each key. **Dupes** allows duplicate key values; **NoCase** makes the key case-insensitive (handy for text keys). Use **+ Add Key** for additional keys.
4. Click **Build Table**.



The builder creates everything the table needs and confirms each piece on screen — the field definitions, the file layout, the data file itself, and a small "map" file that links them together. Once it's done, the table is ready to use.

> Tip: pick an existing table from the **-- Table --** dropdown to load it back into the form and make changes.

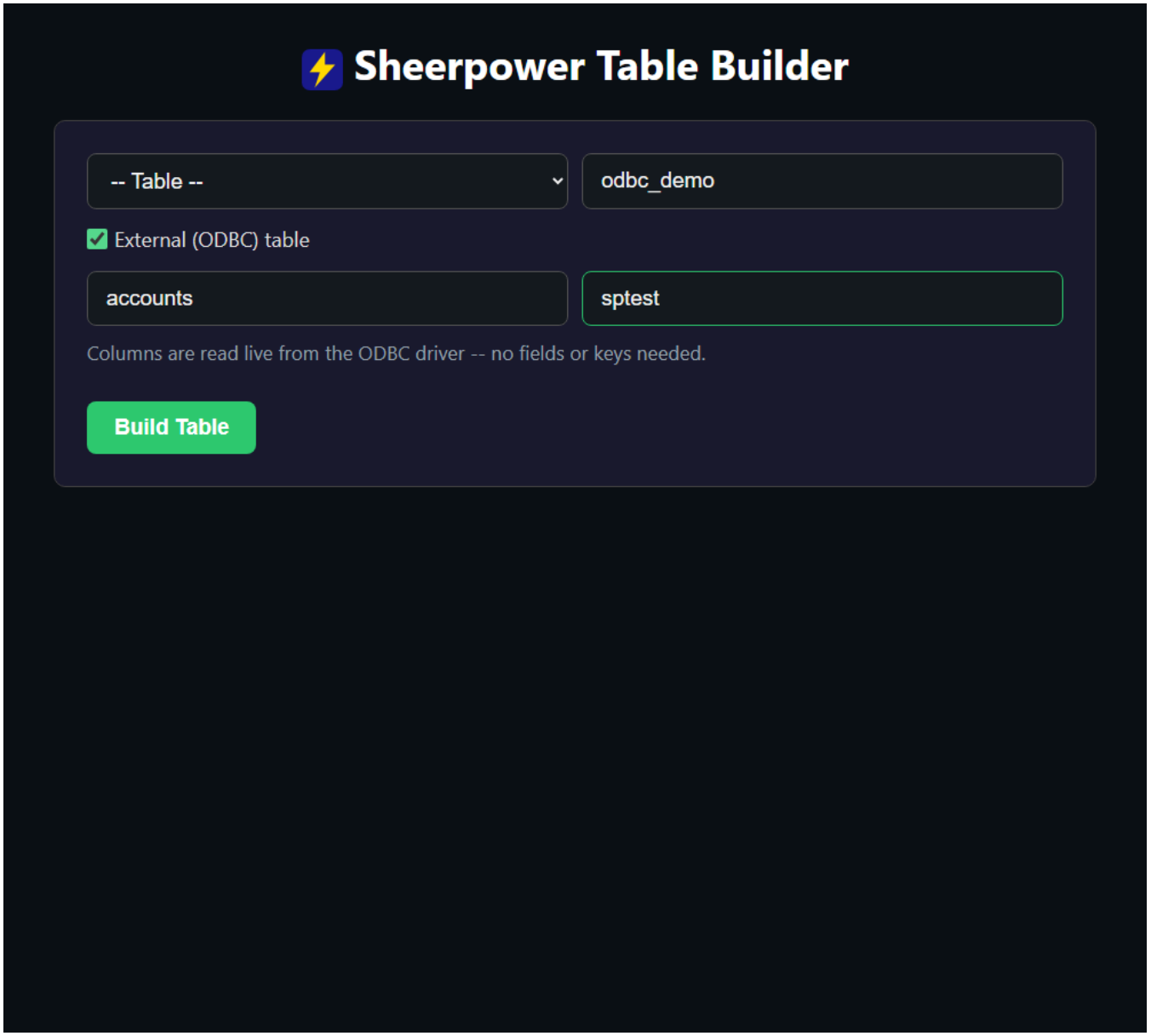
Building an ODBC table reference

Use this when the data already lives in another database and you want to **report on it from Sheerpower** without copying it over. Instead of creating a new data file, the builder simply saves a small pointer to the existing table. Its columns are picked up automatically from the other database, so you don't define any fields or keys yourself.

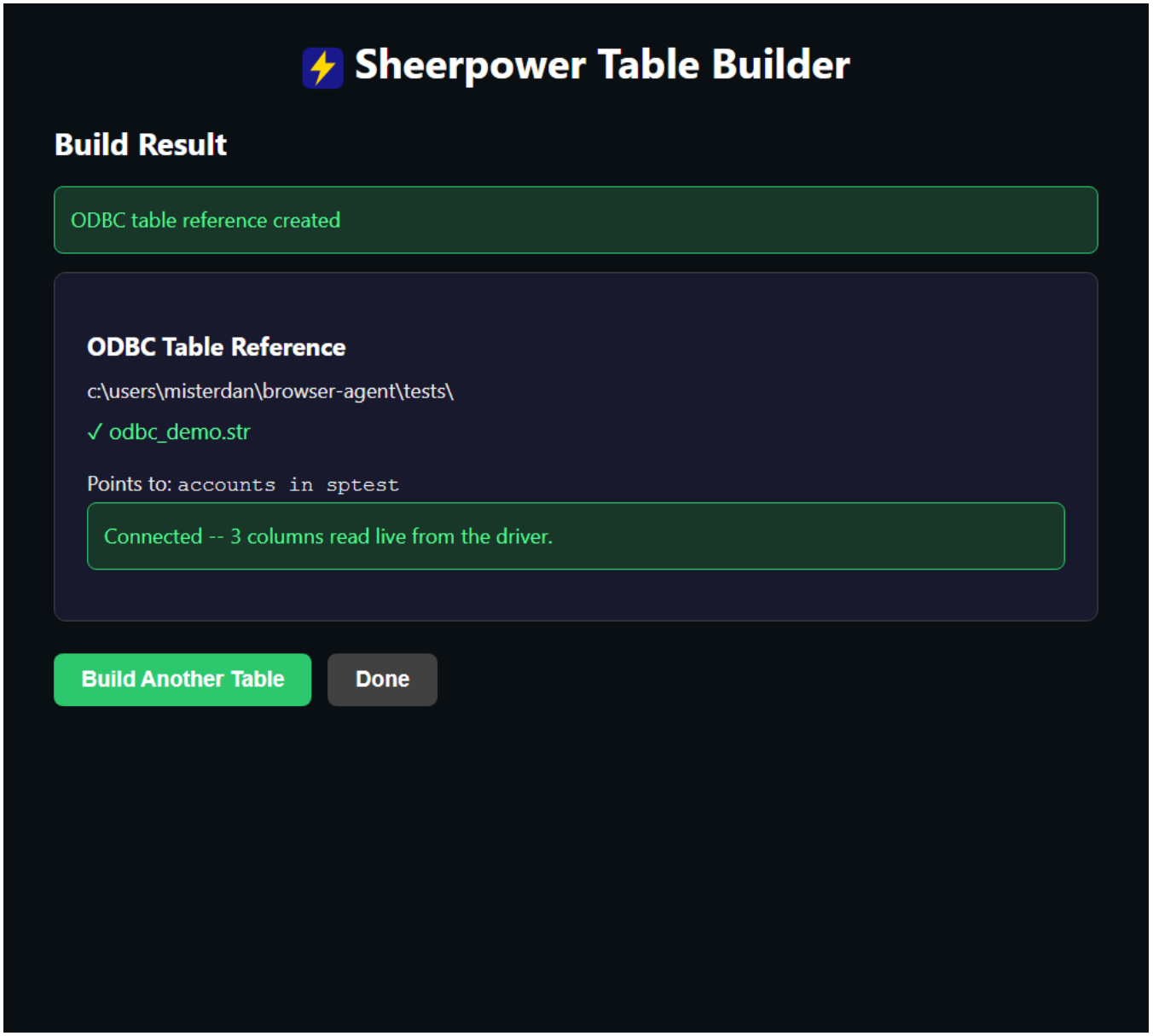
Prerequisite: a User or System **ODBC Data Source (DSN)** must already exist for the target database.

1. Tick **External (ODBC) table**. The Fields and Keys sections disappear — they aren't needed.

2. In the table-name box, enter the **name** you want this reference to have (for example, `odbc_demo`).
3. Enter the **source table name** as it exists in the database, and the **data source (DSN)** that connects to it.



4. Click **Build Table**. The builder saves the reference, then connects to the other database to make sure it works and reports how many columns it found.



If the data source or table name is wrong, the reference is still saved but you get a clear warning ("could not connect — check the data source and source table name") instead of a silent failure.

That's it — the new reference behaves like any other table. Open the Report Writer and you'll find it in the table list, ready to report on, with its numbers and amounts shown and sorted correctly.

Notes and limits

- **Where things are saved:** everything is written to the builder's own folder (shown on the result screen), which is also where other apps look for tables.
- **Naming an ODBC reference:** the table name and data source together can't be longer than 64 characters — the builder will let you know if they are.
- **Column details come from the other database.** For an ODBC reference, the column names, types, and sizes are read automatically; there are no custom display names, prompts, or formatting like a native table has. Numbers and

money amounts still come back exact and are treated as numbers for totals and sorting.

- **Changing an ODBC reference:** just build it again with the same name and it replaces the old one.