NAME

SPI_execute_with_args - execute a command with out-of-line parameters

SYNOPSIS

int SPI_execute_with_args(const char *_command_,
int _nargs_, Oid *_argtypes_,
Datum *_values_, const char *_nulls_,
bool _read_only_, long _count_)

DESCRIPTION

SPI_execute_with_args executes a command that might include references to externally supplied parameters. The command text refers to a parameter as $_n_, and the call specifies data types and values for each such symbol. _read_only_ and _count_ have the same interpretation as in **SPI_execute**.

The main advantage of this routine compared to **SPI_execute** is that data values can be inserted into the command without tedious quoting/escaping, and thus with much less risk of SQL-injection attacks.

Similar results can be achieved with **SPI_prepare** followed by **SPI_execute_plan**; however, when using this function the query plan is always customized to the specific parameter values provided. For one-time query execution, this function should be preferred. If the same command is to be executed with many different parameters, either method might be faster, depending on the cost of re-planning versus the benefit of custom plans.

ARGUMENTS

const char * _command_
command string

int _nargs_
number of input parameters ($1, $2, etc.)

Oid * _argtypes_
an array of length _nargs_, containing the OIDs of the data types of the parameters

Datum * _values_
an array of length _nargs_, containing the actual parameter values

const char * _nulls_
an array of length _nargs_, describing which parameters are null

If _nulls_ is NULL then **SPI_execute_with_args** assumes that no parameters are null. Otherwise,

each entry of the *nulls* array should be ' ' if the corresponding parameter value is non-null, or 'n' if the corresponding parameter value is null. (In the latter case, the actual value in the corresponding *values* entry doesn't matter.) Note that *nulls* is not a text string, just an array: it does not need a '\0' terminator.

bool *read_only*
: true for read-only execution

long *count*
: maximum number of rows to return, or 0 for no limit

RETURN VALUE

The return value is the same as for **SPI_execute**.

SPI_processed and *SPI_tuptable* are set as in **SPI_execute** if successful.