

NAME

`SPI_prepare_extended` - prepare a statement, without executing it yet

SYNOPSIS

```
SPIPlanPtr SPI_prepare_extended(const char * command,
                               const SPIPrepareOptions * options)
```

DESCRIPTION

`SPI_prepare_extended` creates and returns a prepared statement for the specified command, but doesn't execute the command. This function is equivalent to **`SPI_prepare`**, with the addition that the caller can specify options to control the parsing of external parameter references, as well as other facets of query parsing and planning.

ARGUMENTS

const char * *command*
command string

const SPIPrepareOptions * *options*
struct containing optional arguments

Callers should always zero out the entire *options* struct, then fill whichever fields they want to set. This ensures forward compatibility of code, since any fields that are added to the struct in future will be defined to behave backwards-compatibly if they are zero. The currently available *options* fields are:

ParserSetupHook *parserSetup*
Parser hook setup function

void * *parserSetupArg*
pass-through argument for *parserSetup*

RawParseMode *parseMode*
mode for raw parsing; RAW_PARSE_DEFAULT (zero) produces default behavior

int *cursorOptions*
integer bit mask of cursor options; zero produces default behavior

RETURN VALUE

`SPI_prepare_extended` has the same return conventions as **`SPI_prepare`**.