miniSQL Documentation

Release 1.0.0

DevOpSec

May 05, 2019

miniSQL main

class `minisql.`**`MinisqlShell`** (*completekey='tab', stdin=None, stdout=None*)

`do_bye` (*arg*)
Stop recording, close the turtle window, and exit: BYE

`do_circle` (*arg*)
Draw circle with given radius an options extent and steps: CIRCLE 50

`do_color` (*arg*)
Set the color: COLOR BLUE

`do_forward` (*arg*)
Move the turtle forward by the specified distance: FORWARD 10

`do_goto` (*arg*)
Move turtle to an absolute position with changing orientation. GOTO 100 200

`do_heading` (*arg*)
Print the current turtle heading in degrees: HEADING

`do_home` (*arg*)
Return turtle to the home position: HOME

`do_left` (*arg*)
Turn turtle left by given number of degrees: LEFT 90

`do_playback` (*arg*)
Playback commands from a file: PLAYBACK rose.cmd

`do_position` (*arg*)
Print the current turtle position: POSITION

`do_record` (*arg*)
Save future commands to filename: RECORD rose.cmd

`do_reset` (*arg*)
Clear the screen and return turtle to center: RESET

`do_right` (*arg*)
Turn turtle right by given number of degrees: RIGHT 20

`do_undo` (*arg*)
Undo (repeatedly) the last turtle action(s): UNDO

`precmd` (*line*)
Hook method executed just before the command line is interpreted, but after the input prompt is generated and issued.

`minisql.`**`interpretQueries`** (*queries*)
Interpret queries as a list of commands to execute in miniSQL language

Parameters **`queries`** – list of queries to process

Returns False to exit, True to continue (applicable to miniSQL shell)

`minisql.`**`main`** ( )
Parse command line arguments and decide how data is processed

Returns 0 on success, 1 on failure

`minisql.`**`parse`** (*arg*)
Convert a series of zero or more numbers to an argument tuple

`minisql.`**`parseQueries`** (*raw_queries*)
Parse input into interpretable queries based on a SQL-like syntax
Queries are delimited by a semicolon and query commands are delimited by whitespace

Parameters **`raw_queries`** – command string input

Returns list of queries (each as a list of commands)

`minisql.`**`supportsColor`** (*stream*)
Check if terminal supports ASCII color codes

Parameters **`stream`** – file stream to check

Returns True == supported, False == not supported

`minisql.`**`validateStdinReadable`** ( )
Check if stdin has been redirected by parent shell
See the following table for supported input types.
Note that as of python v3.3 symlinks are followed by default.

Stdin Type	Supported	Program Context
directory	no	
keyboard	yes	miniSQL shell
storage	no	
file	yes	miniSQL cmd
symlink	no	
pipe	yes	miniSQL cmd
socket	no	

Returns not readable == -1, 0 == readable in cmd, 1 == readable in shell

CHAPTER 2

Indices and tables

- *Index*
- *Module Index*
- *Search Page*

Python Module Index

Index