:mod:`turtle` --- Turtle graphics

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 5)

Unknown directive type "module".

```
.. module:: turtle
   :synopsis: An educational framework for simple graphics applications
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 8)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Gregor Lingl <gregor.lingl@aon.at>
```

Source code: :source:`Lib/turtle.py`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 10); *backlink*

Unknown interpreted text role "source".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 12)

Unknown directive type "testsetup".

```
.. testsetup:: default

   from turtle import *
   turtle = Turtle()
```

Introduction

Turtle graphics is a popular way for introducing programming to kids. It was part of the original Logo programming language developed by Wally Feurzeig, Seymour Papert and Cynthia Solomon in 1967.

Imagine a robotic turtle starting at (0, 0) in the x-y plane. After an `import turtle`, give it the command `turtle.forward(15)`, and it moves (on-screen!) 15 pixels in the direction it is facing, drawing a line as it moves. Give it the command `turtle.right(25)`, and it rotates in-place 25 degrees clockwise.

By combining together these and similar commands, intricate shapes and pictures can easily be drawn.

The :mod:`turtle` module is an extended reimplementation of the same-named module from the Python standard distribution up to version Python 2.5.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 44); *backlink*

Unknown interpreted text role "mod".

It tries to keep the merits of the old turtle module and to be (nearly) 100% compatible with it. This means in the first place to enable the learning programmer to use all the commands, classes and methods interactively when using the module from within IDLE run with the `-n` switch.

The turtle module provides turtle graphics primitives, in both object-oriented and procedure-oriented ways. Because it uses :mod:`tkinter` for the underlying graphics, it needs a version of Python installed with Tk support.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 52); *backlink*

Unknown interpreted text role "mod".

Turtle star

Turtle can draw intricate shapes using programs that repeat simple moves.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 39)

Unknown directive type "literalinclude".

```
.. literalinclude:: ../includes/turt
```

The object-oriented interface uses essentially two+two classes:

1. The :class:`TurtleScreen` class defines graphics windows as a playground for the drawing turtles. Its constructor needs a :class:`tkinter.Canvas` or a :class:`ScrolledCanvas` as argument. It should be used when :mod:`turtle` is used as part of some application.

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 58**); ***backlink***

   Unknown interpreted text role "class".

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 58**); ***backlink***

   Unknown interpreted text role "class".

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 58**); ***backlink***

   Unknown interpreted text role "class".

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 58**); ***backlink***

   Unknown interpreted text role "mod".

   The function :func:`Screen` returns a singleton object of a :class:`TurtleScreen` subclass. This function should be used when :mod:`turtle` is used as a standalone tool for doing graphics. As a singleton object, inheriting from its class is not possible.

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 63**); ***backlink***

   Unknown interpreted text role "func".

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 63**); ***backlink***

   Unknown interpreted text role "class".

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 63**); ***backlink***

   Unknown interpreted text role "mod".

   All methods of TurtleScreen/Screen also exist as functions, i.e. as part of the procedure-oriented interface.

2. :class:`RawTurtle` (alias: :class:`RawPen`) defines Turtle objects which draw on a :class:`TurtleScreen`. Its constructor needs a Canvas, ScrolledCanvas or TurtleScreen as argument, so the RawTurtle objects know where to draw.

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 71**); ***backlink***

   Unknown interpreted text role "class".

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 71**); ***backlink***

   Unknown interpreted text role "class".

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 71**); ***backlink***

   Unknown interpreted text role "class".

   Derived from RawTurtle is the subclass :class:`Turtle` (alias: :class:`Pen`), which draws on "the" :class:`Screen` instance which is automatically created, if not already present.

   System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-

resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 75); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 75); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 75); *backlink*

Unknown interpreted text role "class".

All methods of RawTurtle/Turtle also exist as functions, i.e. part of the procedure-oriented interface.

The procedural interface provides functions which are derived from the methods of the classes :class:`Screen` and :class:`Turtle`. They have the same names as the corresponding methods. A screen object is automatically created whenever a function derived from a Screen method is called. An (unnamed) turtle object is automatically created whenever any of the functions derived from a Turtle method is called.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 82); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 82); *backlink*

Unknown interpreted text role "class".

To use multiple turtles on a screen one has to use the object-oriented interface.

Note

In the following documentation the argument list for functions is given. Methods, of course, have the additional first argument *self* which is omitted here.

Overview of available Turtle and Screen methods

Turtle methods

Turtle motion

Move and draw

:func:`forward` | :func:`fd`
:func:`backward` | :func:`bk` | :func:`back`
:func:`right` | :func:`rt`
:func:`left` | :func:`lt`
:func:`goto` | :func:`setpos` | :func:`setposition`
:func:`setx`
:func:`sety`
:func:`setheading` | :func:`seth`
:func:`home`
:func:`circle`
:func:`dot`
:func:`stamp`
:func:`clearstamp`
:func:`clearstamps`
:func:`undo`
:func:`speed`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 105); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 105); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 106); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 106); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 106); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 107); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 107); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 108); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 108); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 109); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 109); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 109); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 110); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 111); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 112); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 112**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 113**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 114**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 115**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 116**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 117**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 118**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 119**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 120**); *backlink*

Unknown interpreted text role "func".

Tell Turtle's state

:func:`position` | :func:`pos`
:func:`towards`
:func:`xcor`
:func:`ycor`
:func:`heading`
:func:`distance`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 123**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 123**); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 124); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 125); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 126); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 127); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 128); ***backlink***

Unknown interpreted text role "func".

Setting and measurement

:func:`degrees`
:func:`radians`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 131); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 132); ***backlink***

Unknown interpreted text role "func".

Pen control

Drawing state

:func:`pendown` | :func:`pd` | :func:`down`
:func:`penup` | :func:`pu` | :func:`up`
:func:`pensize` | :func:`width`
:func:`pen`
:func:`isdown`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 136); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 136); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 136); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-`

resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 137); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 137); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 137); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 138); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 138); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 139); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 140); *backlink*

Unknown interpreted text role "func".

Color control

:func:`color`
:func:`pencolor`
:func:`fillcolor`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 143); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 144); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 145); *backlink*

Unknown interpreted text role "func".

Filling

:func:`filling`
:func:`begin_fill`
:func:`end_fill`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 148); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 149); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 150); *backlink*

Unknown interpreted text role "func".

More drawing control

:func:`reset`
:func:`clear`
:func:`write`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 153); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 154); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 155); *backlink*

Unknown interpreted text role "func".

Turtle state

Visibility

:func:`showturtle` | :func:`st`
:func:`hideturtle` | :func:`ht`
:func:`isvisible`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 159); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 159); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 160); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 160); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 161); *backlink*

Unknown interpreted text role "func".

Appearance

:func:`shape`
:func:`resizemode`

:func:`shapesize` | :func:`turtlesize`
:func:`shearfactor`
:func:`settiltangle`
:func:`tiltangle`
:func:`tilt`
:func:`shapetransform`
:func:`get_shapepoly`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 164); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 165); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 166); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 166); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 167); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 168); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 169); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 170); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 171); *backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 172); *backlink***

Unknown interpreted text role "func".

Using events

:func:`onclick`
:func:`onrelease`
:func:`ondrag`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line**

175); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 176); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 177); *backlink*

Unknown interpreted text role "func".

Special Turtle methods

> :func:`begin_poly`
> :func:`end_poly`
> :func:`get_poly`
> :func:`clone`
> :func:`getturtle` | :func:`getpen`
> :func:`getscreen`
> :func:`setundobuffer`
> :func:`undobufferentries`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 180); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 181); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 182); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 183); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 184); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 184); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 185); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 186); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-`

resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 187); *backlink*

Unknown interpreted text role "func".

Methods of TurtleScreen/Screen

Window control

:func:`bgcolor`
:func:`bgpic`
:func:`clearscreen`
:func:`resetscreen`
:func:`screensize`
:func:`setworldcoordinates`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 194); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 195); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 196); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 197); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 198); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 199); *backlink*

Unknown interpreted text role "func".

Animation control

:func:`delay`
:func:`tracer`
:func:`update`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 202); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 203); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 204); *backlink*

Unknown interpreted text role "func".

Using screen events

:func:`listen`
:func:`onkey` | :func:`onkeyrelease`
:func:`onkeypress`
:func:`onclick` | :func:`onscreenclick`
:func:`ontimer`
:func:`mainloop` | :func:`done`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 207**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 208**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 208**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 209**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 210**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 210**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 211**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 212**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 212**); ***backlink***

Unknown interpreted text role "func".

Settings and special methods

:func:`mode`
:func:`colormode`
:func:`getcanvas`
:func:`getshapes`
:func:`register_shape` | :func:`addshape`
:func:`turtles`
:func:`window_height`
:func:`window_width`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, **line 215**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 216**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 217**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 218**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 219**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 219**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 220**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 221**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 222**); ***backlink***

Unknown interpreted text role "func".

Input methods

:func:`textinput`
:func:`numinput`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 225**); ***backlink***

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line 226**); ***backlink***

Unknown interpreted text role "func".

Methods specific to Screen

:func:`bye`
:func:`exitonclick`
:func:`setup`
:func:`title`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, **line**

229); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 230); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 231); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 232); *backlink*

Unknown interpreted text role "func".

Methods of RawTurtle/Turtle and corresponding functions

Most of the examples in this section refer to a Turtle instance called `turtle`.

Turtle motion

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 244)

Unknown directive type "function".

```
.. function:: forward(distance)
              fd(distance)

   :param distance: a number (integer or float)

   Move the turtle forward by the specified *distance*, in the direction the
   turtle is headed.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.position()
      (0.00,0.00)
      >>> turtle.forward(25)
      >>> turtle.position()
      (25.00,0.00)
      >>> turtle.forward(-75)
      >>> turtle.position()
      (-50.00,0.00)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 265)

Unknown directive type "function".

```
.. function:: back(distance)
              bk(distance)
              backward(distance)

   :param distance: a number

   Move the turtle backward by *distance*, opposite to the direction the
   turtle is headed.  Do not change the turtle's heading.

   .. doctest::
      :hide:

      >>> turtle.goto(0, 0)

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.position()
      (0.00,0.00)
      >>> turtle.backward(30)
      >>> turtle.position()
      (-30.00,0.00)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 289)

Unknown directive type "function".

```
.. function:: right(angle)
              rt(angle)

   :param angle: a number (integer or float)

   Turn turtle right by *angle* units.  (Units are by default degrees, but
   can be set via the :func:`degrees` and :func:`radians` functions.)  Angle
   orientation depends on the turtle mode, see :func:`mode`.

   .. doctest::
      :skipif: _tkinter is None
      :hide:

      >>> turtle.setheading(22)

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.heading()
      22.0
      >>> turtle.right(45)
      >>> turtle.heading()
      337.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 314)

Unknown directive type "function".

```
.. function:: left(angle)
              lt(angle)

   :param angle: a number (integer or float)

   Turn turtle left by *angle* units.  (Units are by default degrees, but
   can be set via the :func:`degrees` and :func:`radians` functions.)  Angle
   orientation depends on the turtle mode, see :func:`mode`.

   .. doctest::
      :skipif: _tkinter is None
      :hide:

      >>> turtle.setheading(22)

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.heading()
      22.0
      >>> turtle.left(45)
      >>> turtle.heading()
      67.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 339)

Unknown directive type "function".

```
.. function:: goto(x, y=None)
              setpos(x, y=None)
              setposition(x, y=None)

   :param x: a number or a pair/vector of numbers
   :param y: a number or ``None``

   If *y* is ``None``, *x* must be a pair of coordinates or a :class:`Vec2D`
   (e.g. as returned by :func:`pos`).

   Move turtle to an absolute position.  If the pen is down, draw line.  Do
   not change the turtle's orientation.

   .. doctest::
      :skipif: _tkinter is None
      :hide:

      >>> turtle.goto(0, 0)

   .. doctest::
      :skipif: _tkinter is None

       >>> tp = turtle.pos()
       >>> tp
       (0.00,0.00)
       >>> turtle.setpos(60,30)
       >>> turtle.pos()
       (60.00,30.00)
       >>> turtle.setpos((20,80))
       >>> turtle.pos()
       (20.00,80.00)
       >>> turtle.setpos(tp)
       >>> turtle.pos()
       (0.00,0.00)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 375)

Unknown directive type "function".

```
.. function:: setx(x)

   :param x: a number (integer or float)

   Set the turtle's first coordinate to *x*, leave second coordinate
   unchanged.

   .. doctest::
      :skipif: _tkinter is None
      :hide:

      >>> turtle.goto(0, 240)

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.position()
      (0.00,240.00)
      >>> turtle.setx(10)
      >>> turtle.position()
      (10.00,240.00)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 398)

Unknown directive type "function".

```
.. function:: sety(y)

   :param y: a number (integer or float)

   Set the turtle's second coordinate to *y*, leave first coordinate unchanged.

   .. doctest::
      :skipif: _tkinter is None
      :hide:

      >>> turtle.goto(0, 40)

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.position()
      (0.00,40.00)
      >>> turtle.sety(-10)
      >>> turtle.position()
      (0.00,-10.00)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 420)

Unknown directive type "function".

```
.. function:: setheading(to_angle)
              seth(to_angle)

   :param to_angle: a number (integer or float)

   Set the orientation of the turtle to *to_angle*.  Here are some common
   directions in degrees:

   =================== ====================
    standard mode           logo mode
   =================== ====================
      0 - east                0 - north
     90 - north              90 - east
    180 - west              180 - south
    270 - south             270 - west
   =================== ====================

   .. doctest::
      :skipif: _tkinter is None
```

```
          >>> turtle.setheading(90)
          >>> turtle.heading()
          90.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 445)

Unknown directive type "function".

```
    .. function:: home()

       Move turtle to the origin -- coordinates (0,0) -- and set its heading to
       its start-orientation (which depends on the mode, see :func:`mode`).

       .. doctest::
          :skipif: _tkinter is None
          :hide:

          >>> turtle.setheading(90)
          >>> turtle.goto(0, -10)

       .. doctest::
          :skipif: _tkinter is None

          >>> turtle.heading()
          90.0
          >>> turtle.position()
          (0.00,-10.00)
          >>> turtle.home()
          >>> turtle.position()
          (0.00,0.00)
          >>> turtle.heading()
          0.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 471)

Unknown directive type "function".

```
    .. function:: circle(radius, extent=None, steps=None)

       :param radius: a number
       :param extent: a number (or ``None``)
       :param steps: an integer (or ``None``)

       Draw a circle with given *radius*.  The center is *radius* units left of
       the turtle; *extent* -- an angle -- determines which part of the circle
       is drawn.  If *extent* is not given, draw the entire circle.  If *extent*
       is not a full circle, one endpoint of the arc is the current pen
       position.  Draw the arc in counterclockwise direction if *radius* is
       positive, otherwise in clockwise direction.  Finally the direction of the
       turtle is changed by the amount of *extent*.

       As the circle is approximated by an inscribed regular polygon, *steps*
       determines the number of steps to use.  If not given, it will be
       calculated automatically.  May be used to draw regular polygons.

       .. doctest::
          :skipif: _tkinter is None

          >>> turtle.home()
          >>> turtle.position()
          (0.00,0.00)
          >>> turtle.heading()
          0.0
          >>> turtle.circle(50)
          >>> turtle.position()
          (-0.00,0.00)
          >>> turtle.heading()
          0.0
          >>> turtle.circle(120, 180)  # draw a semicircle
          >>> turtle.position()
          (0.00,240.00)
          >>> turtle.heading()
          180.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 509)

Unknown directive type "function".

```
    .. function:: dot(size=None, *color)

       :param size: an integer >= 1 (if given)
       :param color: a colorstring or a numeric color tuple
```

```
   Draw a circular dot with diameter *size*, using *color*.  If *size* is
   not given, the maximum of pensize+4 and 2*pensize is used.


   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.home()
      >>> turtle.dot()
      >>> turtle.fd(50); turtle.dot(20, "blue"); turtle.fd(50)
      >>> turtle.position()
      (100.00,-0.00)
      >>> turtle.heading()
      0.0
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 530)

Unknown directive type "function".

```
.. function:: stamp()

   Stamp a copy of the turtle shape onto the canvas at the current turtle
   position.  Return a stamp_id for that stamp, which can be used to delete
   it by calling ``clearstamp(stamp_id)``.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.color("blue")
      >>> turtle.stamp()
      11
      >>> turtle.fd(50)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 545)

Unknown directive type "function".

```
.. function:: clearstamp(stampid)

   :param stampid: an integer, must be return value of previous
                   :func:`stamp` call

   Delete stamp with given *stampid*.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.position()
      (150.00,-0.00)
      >>> turtle.color("blue")
      >>> astamp = turtle.stamp()
      >>> turtle.fd(50)
      >>> turtle.position()
      (200.00,-0.00)
      >>> turtle.clearstamp(astamp)
      >>> turtle.position()
      (200.00,-0.00)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 567)

Unknown directive type "function".

```
.. function:: clearstamps(n=None)

   :param n: an integer (or ``None``)

   Delete all or first/last *n* of turtle's stamps.  If *n* is ``None``, delete
   all stamps, if *n* > 0 delete first *n* stamps, else if *n* < 0 delete
   last *n* stamps.

   .. doctest::

      >>> for i in range(8):
      ...     turtle.stamp(); turtle.fd(30)
      13
      14
      15
      16
      17
      18
      19
      20
      >>> turtle.clearstamps(2)
      >>> turtle.clearstamps(-2)
      >>> turtle.clearstamps()
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 592)

Unknown directive type "function".

```
.. function:: undo()

   Undo (repeatedly) the last turtle action(s).  Number of available
   undo actions is determined by the size of the undobuffer.

   .. doctest::
      :skipif: _tkinter is None

      >>> for i in range(4):
      ...     turtle.fd(50); turtle.lt(80)
      ...
      >>> for i in range(8):
      ...     turtle.undo()
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 607)

Unknown directive type "function".

```
.. function:: speed(speed=None)

   :param speed: an integer in the range 0..10 or a speedstring (see below)

   Set the turtle's speed to an integer value in the range 0..10.  If no
   argument is given, return current speed.

   If input is a number greater than 10 or smaller than 0.5, speed is set
   to 0.  Speedstrings are mapped to speedvalues as follows:

   * "fastest":  0
   * "fast":  10
   * "normal":  6
   * "slow":  3
   * "slowest":  1

   Speeds from 1 to 10 enforce increasingly faster animation of line drawing
   and turtle turning.

   Attention: *speed* = 0 means that *no* animation takes
   place. forward/back makes turtle jump and likewise left/right make the
   turtle turn instantly.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.speed()
      3
      >>> turtle.speed('normal')
      >>> turtle.speed()
      6
      >>> turtle.speed(9)
      >>> turtle.speed()
      9
```

Tell Turtle's state

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 646)

Unknown directive type "function".

```
.. function:: position()
              pos()

   Return the turtle's current location (x,y) (as a :class:`Vec2D` vector).

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.pos()
      (440.00,-0.00)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-

main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 658)

Unknown directive type "function".

```
.. function:: towards(x, y=None)

   :param x: a number or a pair/vector of numbers or a turtle instance
   :param y: a number if *x* is a number, else ``None``

   Return the angle between the line from turtle position to position specified
   by (x,y), the vector or the other turtle.  This depends on the turtle's start
   orientation which depends on the mode - "standard"/"world" or "logo".

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.goto(10, 10)
      >>> turtle.towards(0,0)
      225.0
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 675)

Unknown directive type "function".

```
.. function:: xcor()

   Return the turtle's x coordinate.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.home()
      >>> turtle.left(50)
      >>> turtle.forward(100)
      >>> turtle.pos()
      (64.28,76.60)
      >>> print(round(turtle.xcor(), 5))
      64.27876
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 691)

Unknown directive type "function".

```
.. function:: ycor()

   Return the turtle's y coordinate.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.home()
      >>> turtle.left(60)
      >>> turtle.forward(100)
      >>> print(turtle.pos())
      (50.00,86.60)
      >>> print(round(turtle.ycor(), 5))
      86.60254
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 707)

Unknown directive type "function".

```
.. function:: heading()

   Return the turtle's current heading (value depends on the turtle mode, see
   :func:`mode`).

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.home()
      >>> turtle.left(67)
      >>> turtle.heading()
      67.0
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 721)

Unknown directive type "function".

```
.. function:: distance(x, y=None)
```

```
    :param x: a number or a pair/vector of numbers or a turtle instance
    :param y: a number if *x* is a number, else ``None``

    Return the distance from the turtle to (x,y), the given vector, or the given
    other turtle, in turtle step units.

    .. doctest::
       :skipif: _tkinter is None

       >>> turtle.home()
       >>> turtle.distance(30,40)
       50.0
       >>> turtle.distance((30,40))
       50.0
       >>> joe = Turtle()
       >>> joe.forward(77)
       >>> turtle.distance(joe)
       77.0
```

Settings for measurement

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 746)

Unknown directive type "function".

```
.. function:: degrees(fullcircle=360.0)

   :param fullcircle: a number

   Set angle measurement units, i.e. set number of "degrees" for a full circle.
   Default value is 360 degrees.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.home()
      >>> turtle.left(90)
      >>> turtle.heading()
      90.0

      Change angle measurement unit to grad (also known as gon,
      grade, or gradian and equals 1/100-th of the right angle.)
      >>> turtle.degrees(400.0)
      >>> turtle.heading()
      100.0
      >>> turtle.degrees(360)
      >>> turtle.heading()
      90.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 771)

Unknown directive type "function".

```
.. function:: radians()

   Set the angle measurement units to radians.  Equivalent to
   ``degrees(2*math.pi)``.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.home()
      >>> turtle.left(90)
      >>> turtle.heading()
      90.0
      >>> turtle.radians()
      >>> turtle.heading()
      1.5707963267948966

   .. doctest::
      :skipif: _tkinter is None
      :hide:

      >>> turtle.degrees(360)
```

Pen control

Drawing state

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 800)

Unknown directive type "function".

```
.. function:: pendown()
              pd()
              down()

   Pull the pen down -- drawing when moving.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 807)

Unknown directive type "function".

```
.. function:: penup()
              pu()
              up()

   Pull the pen up -- no drawing when moving.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 814)

Unknown directive type "function".

```
.. function:: pensize(width=None)
              width(width=None)

   :param width: a positive number

   Set the line thickness to *width* or return it.  If resizemode is set to
   "auto" and turtleshape is a polygon, that polygon is drawn with the same line
   thickness.  If no argument is given, the current pensize is returned.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.pensize()
      1
      >>> turtle.pensize(10)   # from here on lines of width 10 are drawn
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 831)

Unknown directive type "function".

```
.. function:: pen(pen=None, **pendict)

   :param pen: a dictionary with some or all of the below listed keys
   :param pendict: one or more keyword-arguments with the below listed keys as keywords

   Return or set the pen's attributes in a "pen-dictionary" with the following
   key/value pairs:

   * "shown": True/False
   * "pendown": True/False
   * "pencolor": color-string or color-tuple
   * "fillcolor": color-string or color-tuple
   * "pensize": positive number
   * "speed": number in range 0..10
   * "resizemode": "auto" or "user" or "noresize"
   * "stretchfactor": (positive number, positive number)
   * "outline": positive number
   * "tilt": number

   This dictionary can be used as argument for a subsequent call to :func:`pen`
   to restore the former pen-state.  Moreover one or more of these attributes
   can be provided as keyword-arguments.  This can be used to set several pen
   attributes in one statement.

   .. doctest::
      :skipif: _tkinter is None
      :options: +NORMALIZE_WHITESPACE

      >>> turtle.pen(fillcolor="black", pencolor="red", pensize=10)
      >>> sorted(turtle.pen().items())
      [('fillcolor', 'black'), ('outline', 1), ('pencolor', 'red'),
       ('pendown', True), ('pensize', 10), ('resizemode', 'noresize'),
       ('shearfactor', 0.0), ('shown', True), ('speed', 9),
       ('stretchfactor', (1.0, 1.0)), ('tilt', 0.0)]
      >>> penstate=turtle.pen()
      >>> turtle.color("yellow", "")
      >>> turtle.penup()
      >>> sorted(turtle.pen().items())[:3]
      [('fillcolor', ''), ('outline', 1), ('pencolor', 'yellow')]
      >>> turtle.pen(penstate, fillcolor="green")
      >>> sorted(turtle.pen().items())[:3]
```

```
        [('fillcolor', 'green'), ('outline', 1), ('pencolor', 'red')]
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 874)

Unknown directive type "function".

```
.. function:: isdown()

   Return ``True`` if pen is down, ``False`` if it's up.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.penup()
      >>> turtle.isdown()
      False
      >>> turtle.pendown()
      >>> turtle.isdown()
      True
```

Color control

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 892)

Unknown directive type "function".

```
.. function:: pencolor(*args)

   Return or set the pencolor.

   Four input formats are allowed:

   ``pencolor()``
      Return the current pencolor as color specification string or
      as a tuple (see example).  May be used as input to another
      color/pencolor/fillcolor call.

   ``pencolor(colorstring)``
      Set pencolor to *colorstring*, which is a Tk color specification string,
      such as ``"red"``, ``"yellow"``, or ``"#33cc8c"``.

   ``pencolor((r, g, b))``
      Set pencolor to the RGB color represented by the tuple of *r*, *g*, and
      *b*.  Each of *r*, *g*, and *b* must be in the range 0..colormode, where
      colormode is either 1.0 or 255 (see :func:`colormode`).

   ``pencolor(r, g, b)``
      Set pencolor to the RGB color represented by *r*, *g*, and *b*.  Each of
      *r*, *g*, and *b* must be in the range 0..colormode.

   If turtleshape is a polygon, the outline of that polygon is drawn with the
   newly set pencolor.

   .. doctest::
      :skipif: _tkinter is None

       >>> colormode()
       1.0
       >>> turtle.pencolor()
       'red'
       >>> turtle.pencolor("brown")
       >>> turtle.pencolor()
       'brown'
       >>> tup = (0.2, 0.8, 0.55)
       >>> turtle.pencolor(tup)
       >>> turtle.pencolor()
       (0.2, 0.8, 0.5490196078431373)
       >>> colormode(255)
       >>> turtle.pencolor()
       (51.0, 204.0, 140.0)
       >>> turtle.pencolor('#32c18f')
       >>> turtle.pencolor()
       (50.0, 193.0, 143.0)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 941)

Unknown directive type "function".

```
.. function:: fillcolor(*args)

   Return or set the fillcolor.

   Four input formats are allowed:
```

```
   ``fillcolor()``
      Return the current fillcolor as color specification string, possibly
      in tuple format (see example).  May be used as input to another
      color/pencolor/fillcolor call.

   ``fillcolor(colorstring)``
      Set fillcolor to *colorstring*, which is a Tk color specification string,
      such as ``"red"``, ``"yellow"``, or ``"#33cc8c"``.

   ``fillcolor((r, g, b))``
      Set fillcolor to the RGB color represented by the tuple of *r*, *g*, and
      *b*.  Each of *r*, *g*, and *b* must be in the range 0..colormode, where
      colormode is either 1.0 or 255 (see :func:`colormode`).

   ``fillcolor(r, g, b)``
      Set fillcolor to the RGB color represented by *r*, *g*, and *b*.  Each of
      *r*, *g*, and *b* must be in the range 0..colormode.

   If turtleshape is a polygon, the interior of that polygon is drawn
   with the newly set fillcolor.

   .. doctest::
      :skipif: _tkinter is None

       >>> turtle.fillcolor("violet")
       >>> turtle.fillcolor()
       'violet'
       >>> turtle.pencolor()
       (50.0, 193.0, 143.0)
       >>> turtle.fillcolor((50, 193, 143))  # Integers, not floats
       >>> turtle.fillcolor()
       (50.0, 193.0, 143.0)
       >>> turtle.fillcolor('#ffffff')
       >>> turtle.fillcolor()
       (255.0, 255.0, 255.0)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 984)

Unknown directive type "function".

```
.. function:: color(*args)

   Return or set pencolor and fillcolor.

   Several input formats are allowed.  They use 0 to 3 arguments as
   follows:

   ``color()``
      Return the current pencolor and the current fillcolor as a pair of color
      specification strings or tuples as returned by :func:`pencolor` and
      :func:`fillcolor`.

   ``color(colorstring)``, ``color((r,g,b))``, ``color(r,g,b)``
      Inputs as in :func:`pencolor`, set both, fillcolor and pencolor, to the
      given value.

   ``color(colorstring1, colorstring2)``, ``color((r1,g1,b1), (r2,g2,b2))``
      Equivalent to ``pencolor(colorstring1)`` and ``fillcolor(colorstring2)``
      and analogously if the other input format is used.

   If turtleshape is a polygon, outline and interior of that polygon is drawn
   with the newly set colors.

   .. doctest::
      :skipif: _tkinter is None

       >>> turtle.color("red", "green")
       >>> turtle.color()
       ('red', 'green')
       >>> color("#285078", "#a0c8f0")
       >>> color()
       ((40.0, 80.0, 120.0), (160.0, 200.0, 240.0))
```

See also: Screen method :func:`colormode`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1018); *backlink*

Unknown interpreted text role "func".

Filling

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1024)

Unknown directive type "doctest".

```
.. doctest::
   :skipif: _tkinter is None
   :hide:

   >>> turtle.home()
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1030)

Unknown directive type "function".

```
.. function:: filling()

   Return fillstate (``True`` if filling, ``False`` else).

   .. doctest::
      :skipif: _tkinter is None

       >>> turtle.begin_fill()
       >>> if turtle.filling():
       ...    turtle.pensize(5)
       ... else:
       ...    turtle.pensize(3)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1045)

Unknown directive type "function".

```
.. function:: begin_fill()

   To be called just before drawing a shape to be filled.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1050)

Unknown directive type "function".

```
.. function:: end_fill()

   Fill the shape drawn after the last call to :func:`begin_fill`.

   Whether or not overlap regions for self-intersecting polygons
   or multiple shapes are filled depends on the operating system graphics,
   type of overlap, and number of overlaps.  For example, the Turtle star
   above may be either all yellow or have some white regions.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.color("black", "red")
      >>> turtle.begin_fill()
      >>> turtle.circle(80)
      >>> turtle.end_fill()
```

More drawing control

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1071)

Unknown directive type "function".

```
.. function:: reset()

   Delete the turtle's drawings from the screen, re-center the turtle and set
   variables to the default values.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.goto(0,-22)
      >>> turtle.left(100)
      >>> turtle.position()
      (0.00,-22.00)
      >>> turtle.heading()
      100.0
      >>> turtle.reset()
      >>> turtle.position()
      (0.00,0.00)
      >>> turtle.heading()
      0.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1092)

Unknown directive type "function".

```
.. function:: clear()

   Delete the turtle's drawings from the screen.  Do not move turtle.  State and
   position of the turtle as well as drawings of other turtles are not affected.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1098)

Unknown directive type "function".

```
.. function:: write(arg, move=False, align="left", font=("Arial", 8, "normal"))

   :param arg: object to be written to the TurtleScreen
   :param move: True/False
   :param align: one of the strings "left", "center" or right"
   :param font: a triple (fontname, fontsize, fonttype)

   Write text - the string representation of *arg* - at the current turtle
   position according to *align* ("left", "center" or "right") and with the given
   font.  If *move* is true, the pen is moved to the bottom-right corner of the
   text.  By default, *move* is ``False``.

   >>> turtle.write("Home = ", True, align="center")
   >>> turtle.write((0,0), True)
```

Turtle state

Visibility

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1120)

Unknown directive type "function".

```
.. function:: hideturtle()
              ht()

   Make the turtle invisible.  It's a good idea to do this while you're in the
   middle of doing some complex drawing, because hiding the turtle speeds up the
   drawing observably.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.hideturtle()
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1133)

Unknown directive type "function".

```
.. function:: showturtle()
              st()

   Make the turtle visible.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.showturtle()
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1144)

Unknown directive type "function".

```
.. function:: isvisible()

   Return ``True`` if the Turtle is shown, ``False`` if it's hidden.

   >>> turtle.hideturtle()
   >>> turtle.isvisible()
   False
   >>> turtle.showturtle()
   >>> turtle.isvisible()
```

```
      True
```

Appearance

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1159)

Unknown directive type "function".

```
.. function:: shape(name=None)

   :param name: a string which is a valid shapename

   Set turtle shape to shape with given *name* or, if name is not given, return
   name of current shape.  Shape with *name* must exist in the TurtleScreen's
   shape dictionary.  Initially there are the following polygon shapes: "arrow",
   "turtle", "circle", "square", "triangle", "classic".  To learn about how to
   deal with shapes see Screen method :func:`register_shape`.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.shape()
      'classic'
      >>> turtle.shape("turtle")
      >>> turtle.shape()
      'turtle'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1179)

Unknown directive type "function".

```
.. function:: resizemode(rmode=None)

   :param rmode: one of the strings "auto", "user", "noresize"

   Set resizemode to one of the values: "auto", "user", "noresize".  If *rmode*
   is not given, return current resizemode.  Different resizemodes have the
   following effects:

   - "auto": adapts the appearance of the turtle corresponding to the value of pensize.
   - "user": adapts the appearance of the turtle according to the values of
     stretchfactor and outlinewidth (outline), which are set by
     :func:`shapesize`.
   - "noresize": no adaption of the turtle's appearance takes place.

   ``resizemode("user")`` is called by :func:`shapesize` when used with arguments.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.resizemode()
      'noresize'
      >>> turtle.resizemode("auto")
      >>> turtle.resizemode()
      'auto'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1205)

Unknown directive type "function".

```
.. function:: shapesize(stretch_wid=None, stretch_len=None, outline=None)
              turtlesize(stretch_wid=None, stretch_len=None, outline=None)

   :param stretch_wid: positive number
   :param stretch_len: positive number
   :param outline: positive number

   Return or set the pen's attributes x/y-stretchfactors and/or outline.  Set
   resizemode to "user".  If and only if resizemode is set to "user", the turtle
   will be displayed stretched according to its stretchfactors: *stretch_wid* is
   stretchfactor perpendicular to its orientation, *stretch_len* is
   stretchfactor in direction of its orientation, *outline* determines the width
   of the shapes's outline.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.shapesize()
      (1.0, 1.0, 1)
      >>> turtle.resizemode("user")
      >>> turtle.shapesize(5, 5, 12)
      >>> turtle.shapesize()
      (5, 5, 12)
      >>> turtle.shapesize(outline=8)
      >>> turtle.shapesize()
      (5, 5, 8)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1233)

Unknown directive type "function".

```
.. function:: shearfactor(shear=None)

   :param shear: number (optional)

   Set or return the current shearfactor. Shear the turtleshape according to
   the given shearfactor shear, which is the tangent of the shear angle.
   Do *not* change the turtle's heading (direction of movement).
   If shear is not given: return the current shearfactor, i. e. the
   tangent of the shear angle, by which lines parallel to the
   heading of the turtle are sheared.

   .. doctest::
      :skipif: _tkinter is None

       >>> turtle.shape("circle")
       >>> turtle.shapesize(5,2)
       >>> turtle.shearfactor(0.5)
       >>> turtle.shearfactor()
       0.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1254)

Unknown directive type "function".

```
.. function:: tilt(angle)

   :param angle: a number

   Rotate the turtleshape by *angle* from its current tilt-angle, but do *not*
   change the turtle's heading (direction of movement).

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.reset()
      >>> turtle.shape("circle")
      >>> turtle.shapesize(5,2)
      >>> turtle.tilt(30)
      >>> turtle.fd(50)
      >>> turtle.tilt(30)
      >>> turtle.fd(50)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1273)

Unknown directive type "function".

```
.. function:: settiltangle(angle)

   :param angle: a number

   Rotate the turtleshape to point in the direction specified by *angle*,
   regardless of its current tilt-angle.  *Do not* change the turtle's heading
   (direction of movement).

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.reset()
      >>> turtle.shape("circle")
      >>> turtle.shapesize(5,2)
      >>> turtle.settiltangle(45)
      >>> turtle.fd(50)
      >>> turtle.settiltangle(-45)
      >>> turtle.fd(50)

   .. deprecated:: 3.1
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1295)

Unknown directive type "function".

```
.. function:: tiltangle(angle=None)

   :param angle: a number (optional)

   Set or return the current tilt-angle. If angle is given, rotate the
   turtleshape to point in the direction specified by angle,
   regardless of its current tilt-angle. Do *not* change the turtle's
   heading (direction of movement).
   If angle is not given: return the current tilt-angle, i. e. the angle
   between the orientation of the turtleshape and the heading of the
   turtle (its direction of movement).

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.reset()
      >>> turtle.shape("circle")
      >>> turtle.shapesize(5,2)
      >>> turtle.tilt(45)
      >>> turtle.tiltangle()
      45.0
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1318)

Unknown directive type "function".

```
.. function:: shapetransform(t11=None, t12=None, t21=None, t22=None)

   :param t11: a number (optional)
   :param t12: a number (optional)
   :param t21: a number (optional)
   :param t12: a number (optional)

   Set or return the current transformation matrix of the turtle shape.

   If none of the matrix elements are given, return the transformation
   matrix as a tuple of 4 elements.
   Otherwise set the given elements and transform the turtleshape
   according to the matrix consisting of first row t11, t12 and
   second row t21, t22. The determinant t11 * t22 - t12 * t21 must not be
   zero, otherwise an error is raised.
   Modify stretchfactor, shearfactor and tiltangle according to the
   given matrix.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle = Turtle()
      >>> turtle.shape("square")
      >>> turtle.shapesize(4,2)
      >>> turtle.shearfactor(-0.5)
      >>> turtle.shapetransform()
      (4.0, -1.0, -0.0, 2.0)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1347)

Unknown directive type "function".

```
.. function:: get_shapepoly()

   Return the current shape polygon as tuple of coordinate pairs. This
   can be used to define a new shape or components of a compound shape.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.shape("square")
      >>> turtle.shapetransform(4, -1, 0, 2)
      >>> turtle.get_shapepoly()
      ((50, -20), (30, 20), (-50, 20), (-30, -20))
```

Using events

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1364)

Unknown directive type "function".

```
.. function:: onclick(fun, btn=1, add=None)
   :noindex:

   :param fun: a function with two arguments which will be called with the
               coordinates of the clicked point on the canvas
   :param btn: number of the mouse-button, defaults to 1 (left mouse button)
   :param add: ``True`` or ``False`` -- if ``True``, a new binding will be
               added, otherwise it will replace a former binding

   Bind *fun* to mouse-click events on this turtle.  If *fun* is ``None``,
   existing bindings are removed.  Example for the anonymous turtle, i.e. the
   procedural way:

   .. doctest::
      :skipif: _tkinter is None

      >>> def turn(x, y):
      ...     left(180)
      ...
      >>> onclick(turn)  # Now clicking into the turtle will turn it.
      >>> onclick(None)  # event-binding will be removed
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1387)

Unknown directive type "function".

```
.. function:: onrelease(fun, btn=1, add=None)

   :param fun: a function with two arguments which will be called with the
               coordinates of the clicked point on the canvas
   :param btn: number of the mouse-button, defaults to 1 (left mouse button)
   :param add: ``True`` or ``False`` -- if ``True``, a new binding will be
               added, otherwise it will replace a former binding

   Bind *fun* to mouse-button-release events on this turtle.  If *fun* is
   ``None``, existing bindings are removed.

   .. doctest::
      :skipif: _tkinter is None

      >>> class MyTurtle(Turtle):
      ...     def glow(self,x,y):
      ...         self.fillcolor("red")
      ...     def unglow(self,x,y):
      ...         self.fillcolor("")
      ...
      >>> turtle = MyTurtle()
      >>> turtle.onclick(turtle.glow)     # clicking on turtle turns fillcolor red,
      >>> turtle.onrelease(turtle.unglow) # releasing turns it to transparent.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1412)

Unknown directive type "function".

```
.. function:: ondrag(fun, btn=1, add=None)

   :param fun: a function with two arguments which will be called with the
               coordinates of the clicked point on the canvas
   :param btn: number of the mouse-button, defaults to 1 (left mouse button)
   :param add: ``True`` or ``False`` -- if ``True``, a new binding will be
               added, otherwise it will replace a former binding

   Bind *fun* to mouse-move events on this turtle.  If *fun* is ``None``,
   existing bindings are removed.

   Remark: Every sequence of mouse-move-events on a turtle is preceded by a
   mouse-click event on that turtle.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.ondrag(turtle.goto)

   Subsequently, clicking and dragging the Turtle will move it across
   the screen thereby producing handdrawings (if pen is down).
```

Special Turtle methods

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1438)

Unknown directive type "function".

```
.. function:: begin_poly()

   Start recording the vertices of a polygon.  Current turtle position is first
   vertex of polygon.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1444)

Unknown directive type "function".

```
.. function:: end_poly()

   Stop recording the vertices of a polygon.  Current turtle position is last
   vertex of polygon.  This will be connected with the first vertex.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1450)

Unknown directive type "function".

```
.. function:: get_poly()

   Return the last recorded polygon.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.home()
      >>> turtle.begin_poly()
      >>> turtle.fd(100)
      >>> turtle.left(20)
      >>> turtle.fd(30)
      >>> turtle.left(60)
      >>> turtle.fd(50)
      >>> turtle.end_poly()
      >>> p = turtle.get_poly()
      >>> register_shape("myFavouriteShape", p)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1469)

Unknown directive type "function".

```
.. function:: clone()

   Create and return a clone of the turtle with same position, heading and
   turtle properties.

   .. doctest::
      :skipif: _tkinter is None

      >>> mick = Turtle()
      >>> joe = mick.clone()
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1481)

Unknown directive type "function".

```
.. function:: getturtle()
              getpen()

   Return the Turtle object itself.  Only reasonable use: as a function to
   return the "anonymous turtle":

   .. doctest::
      :skipif: _tkinter is None

      >>> pet = getturtle()
      >>> pet.fd(50)
      >>> pet
      <turtle.Turtle object at 0x...>
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1496)

Unknown directive type "function".

```
.. function:: getscreen()

   Return the :class:`TurtleScreen` object the turtle is drawing on.
   TurtleScreen methods can then be called for that object.

   .. doctest::
      :skipif: _tkinter is None

      >>> ts = turtle.getscreen()
      >>> ts
```

```
    <turtle._Screen object at 0x...>
    >>> ts.bgcolor("pink")
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1510)

Unknown directive type "function".

```
.. function:: setundobuffer(size)

   :param size: an integer or ``None``

   Set or disable undobuffer.  If *size* is an integer, an empty undobuffer of
   given size is installed.  *size* gives the maximum number of turtle actions
   that can be undone by the :func:`undo` method/function.  If *size* is
   ``None``, the undobuffer is disabled.

   .. doctest::
      :skipif: _tkinter is None

      >>> turtle.setundobuffer(42)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1525)

Unknown directive type "function".

```
.. function:: undobufferentries()

   Return number of entries in the undobuffer.

   .. doctest::
      :skipif: _tkinter is None

      >>> while undobufferentries():
      ...     undo()
```

Compound shapes

To use compound turtle shapes, which consist of several polygons of different color, you must use the helper class :class:`Shape` explicitly as described below:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1542); *backlink*

Unknown interpreted text role "class".

1. Create an empty Shape object of type "compound".
2. Add as many components to this object as desired, using the :meth:`addcomponent` method.

   System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1547); *backlink*

   Unknown interpreted text role "meth".

   For example:

   System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1552)

   Unknown directive type "doctest".

   ```
   .. doctest::
      :skipif: _tkinter is None

      >>> s = Shape("compound")
      >>> poly1 = ((0,0),(10,-5),(0,10),(-10,-5))
      >>> s.addcomponent(poly1, "red", "blue")
      >>> poly2 = ((0,0),(10,-5),(-10,-5))
      >>> s.addcomponent(poly2, "blue", "red")
   ```

3. Now add the Shape to the Screen's shapelist and use it:

   System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line

1563)

Unknown directive type "doctest".

```
.. doctest::
   :skipif: _tkinter is None

   >>> register_shape("myshape", s)
   >>> shape("myshape")
```

Note

The :class:`Shape` class is used internally by the :func:`register_shape` method in different ways. The application programmer has to deal with the Shape class *only* when using compound shapes like shown above!

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1572); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1572); *backlink*

Unknown interpreted text role "func".

Methods of TurtleScreen/Screen and corresponding functions

Most of the examples in this section refer to a TurtleScreen instance called `screen`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1583)

Unknown directive type "doctest".

```
.. doctest::
   :skipif: _tkinter is None
   :hide:

   >>> screen = Screen()
```

Window control

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1592)

Unknown directive type "function".

```
.. function:: bgcolor(*args)

   :param args: a color string or three numbers in the range 0..colormode or a
                3-tuple of such numbers


   Set or return background color of the TurtleScreen.

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.bgcolor("orange")
      >>> screen.bgcolor()
      'orange'
      >>> screen.bgcolor("#800080")
      >>> screen.bgcolor()
      (128.0, 0.0, 128.0)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 1611)

Unknown directive type "function".

```
.. function:: bgpic(picname=None)

   :param picname: a string, name of a gif-file or ``"nopic"``, or ``None``

   Set background image or return name of current backgroundimage.  If *picname*
   is a filename, set the corresponding image as background.  If *picname* is
   ``"nopic"``, delete background image, if present.  If *picname* is ``None``,
   return the filename of the current backgroundimage. ::
```

```
>>> screen.bgpic()
'nopic'
>>> screen.bgpic("landscape.gif")
>>> screen.bgpic()
"landscape.gif"
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1627)

Unknown directive type "function".

```
.. function:: clear()
   :noindex:

   .. note::
      This TurtleScreen method is available as a global function only under the
      name ``clearscreen``.  The global function ``clear`` is a different one
      derived from the Turtle method ``clear``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1636)

Unknown directive type "function".

```
.. function:: clearscreen()

   Delete all drawings and all turtles from the TurtleScreen.  Reset the now
   empty TurtleScreen to its initial state: white background, no background
   image, no event bindings and tracing on.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1643)

Unknown directive type "function".

```
.. function:: reset()
   :noindex:

   .. note::
      This TurtleScreen method is available as a global function only under the
      name ``resetscreen``.  The global function ``reset`` is another one
      derived from the Turtle method ``reset``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1652)

Unknown directive type "function".

```
.. function:: resetscreen()

   Reset all Turtles on the Screen to their initial state.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1657)

Unknown directive type "function".

```
.. function:: screensize(canvwidth=None, canvheight=None, bg=None)

   :param canvwidth: positive integer, new width of canvas in pixels
   :param canvheight: positive integer, new height of canvas in pixels
   :param bg: colorstring or color-tuple, new background color

   If no arguments are given, return current (canvaswidth, canvasheight).  Else
   resize the canvas the turtles are drawing on.  Do not alter the drawing
   window.  To observe hidden parts of the canvas, use the scrollbars. With this
   method, one can make visible those parts of a drawing which were outside the
   canvas before.

      >>> screen.screensize()
      (400, 300)
      >>> screen.screensize(2000,1500)
      >>> screen.screensize()
      (2000, 1500)

   e.g. to search for an erroneously escaped turtle ;-)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1678)

Unknown directive type "function".

```
.. function:: setworldcoordinates(llx, lly, urx, ury)

   :param llx: a number, x-coordinate of lower left corner of canvas
   :param lly: a number, y-coordinate of lower left corner of canvas
   :param urx: a number, x-coordinate of upper right corner of canvas
   :param ury: a number, y-coordinate of upper right corner of canvas

   Set up user-defined coordinate system and switch to mode "world" if
   necessary.  This performs a ``screen.reset()``.  If mode "world" is already
   active, all drawings are redrawn according to the new coordinates.

   **ATTENTION**: in user-defined coordinate systems angles may appear
   distorted.

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.reset()
      >>> screen.setworldcoordinates(-50,-7.5,50,7.5)
      >>> for _ in range(72):
      ...     left(10)
      ...
      >>> for _ in range(8):
      ...     left(45); fd(2)   # a regular octagon

   .. doctest::
      :skipif: _tkinter is None
      :hide:

      >>> screen.reset()
      >>> for t in turtles():
      ...      t.reset()
```

Animation control

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1715)

Unknown directive type "function".

```
.. function:: delay(delay=None)

   :param delay: positive integer

   Set or return the drawing *delay* in milliseconds.  (This is approximately
   the time interval between two consecutive canvas updates.)  The longer the
   drawing delay, the slower the animation.

   Optional argument:

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.delay()
      10
      >>> screen.delay(5)
      >>> screen.delay()
      5
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1735)

Unknown directive type "function".

```
.. function:: tracer(n=None, delay=None)

   :param n: nonnegative integer
   :param delay: nonnegative integer

   Turn turtle animation on/off and set delay for update drawings.  If
   *n* is given, only each n-th regular screen update is really
   performed.  (Can be used to accelerate the drawing of complex
   graphics.)  When called without arguments, returns the currently
   stored value of n. Second argument sets delay value (see
   :func:`delay`).

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.tracer(8, 25)
      >>> dist = 2
      >>> for i in range(200):
      ...     fd(dist)
      ...     rt(90)
...     dist += 2
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1758)

Unknown directive type "function".

```
.. function:: update()

   Perform a TurtleScreen update. To be used when tracer is turned off.
```

See also the RawTurtle/Turtle method :func:`speed`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1762); *backlink*

Unknown interpreted text role "func".

Using screen events

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1768)

Unknown directive type "function".

```
.. function:: listen(xdummy=None, ydummy=None)

   Set focus on TurtleScreen (in order to collect key-events).  Dummy arguments
   are provided in order to be able to pass :func:`listen` to the onclick method.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1774)

Unknown directive type "function".

```
.. function:: onkey(fun, key)
              onkeyrelease(fun, key)

   :param fun: a function with no arguments or ``None``
   :param key: a string: key (e.g. "a") or key-symbol (e.g. "space")

   Bind *fun* to key-release event of key.  If *fun* is ``None``, event bindings
   are removed. Remark: in order to be able to register key-events, TurtleScreen
   must have the focus. (See method :func:`listen`.)

   .. doctest::
      :skipif: _tkinter is None

      >>> def f():
      ...     fd(50)
      ...     lt(60)
      ...
      >>> screen.onkey(f, "Up")
      >>> screen.listen()
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1795)

Unknown directive type "function".

```
.. function:: onkeypress(fun, key=None)

   :param fun: a function with no arguments or ``None``
   :param key: a string: key (e.g. "a") or key-symbol (e.g. "space")

   Bind *fun* to key-press event of key if key is given,
   or to any key-press-event if no key is given.
   Remark: in order to be able to register key-events, TurtleScreen
   must have focus. (See method :func:`listen`.)

   .. doctest::
      :skipif: _tkinter is None

      >>> def f():
      ...     fd(50)
      ...
      >>> screen.onkey(f, "Up")
      >>> screen.listen()
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1815)

Unknown directive type "function".

```
.. function:: onclick(fun, btn=1, add=None)
              onscreenclick(fun, btn=1, add=None)

   :param fun: a function with two arguments which will be called with the
               coordinates of the clicked point on the canvas
   :param btn: number of the mouse-button, defaults to 1 (left mouse button)
   :param add: ``True`` or ``False`` -- if ``True``, a new binding will be
               added, otherwise it will replace a former binding

   Bind *fun* to mouse-click events on this screen.  If *fun* is ``None``,
   existing bindings are removed.

   Example for a TurtleScreen instance named ``screen`` and a Turtle instance
   named ``turtle``:

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.onclick(turtle.goto) # Subsequently clicking into the TurtleScreen will
      >>>                             # make the turtle move to the clicked point.
      >>> screen.onclick(None)        # remove event binding again

   .. note::
      This TurtleScreen method is available as a global function only under the
      name ``onscreenclick``.  The global function ``onclick`` is another one
      derived from the Turtle method ``onclick``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1843)

Unknown directive type "function".

```
.. function:: ontimer(fun, t=0)

   :param fun: a function with no arguments
   :param t: a number >= 0

   Install a timer that calls *fun* after *t* milliseconds.

   .. doctest::
      :skipif: _tkinter is None

      >>> running = True
      >>> def f():
      ...     if running:
      ...         fd(50)
      ...         lt(60)
      ...         screen.ontimer(f, 250)
      >>> f()   ### makes the turtle march around
      >>> running = False
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1863)

Unknown directive type "function".

```
.. function:: mainloop()
              done()

   Starts event loop - calling Tkinter's mainloop function.
   Must be the last statement in a turtle graphics program.
   Must *not* be used if a script is run from within IDLE in -n mode
   (No subprocess) - for interactive use of turtle graphics. ::

      >>> screen.mainloop()
```

Input methods

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 1877)

Unknown directive type "function".

```
.. function:: textinput(title, prompt)

   :param title: string
   :param prompt: string

   Pop up a dialog window for input of a string. Parameter title is
   the title of the dialog window, prompt is a text mostly describing
   what information to input.
   Return the string input. If the dialog is canceled, return ``None``. ::

      >>> screen.textinput("NIM", "Name of first player:")
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1890)

Unknown directive type "function".

```
.. function:: numinput(title, prompt, default=None, minval=None, maxval=None)

   :param title: string
   :param prompt: string
   :param default: number (optional)
   :param minval: number (optional)
   :param maxval: number (optional)

   Pop up a dialog window for input of a number. title is the title of the
   dialog window, prompt is a text mostly describing what numerical information
   to input. default: default value, minval: minimum value for input,
   maxval: maximum value for input.
   The number input must be in the range minval .. maxval if these are
   given. If not, a hint is issued and the dialog remains open for
   correction.
   Return the number input. If the dialog is canceled,  return ``None``. ::

      >>> screen.numinput("Poker", "Your stakes:", 1000, minval=10, maxval=10000)
```

Settings and special methods

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1913)

Unknown directive type "function".

```
.. function:: mode(mode=None)

   :param mode: one of the strings "standard", "logo" or "world"

   Set turtle mode ("standard", "logo" or "world") and perform reset.  If mode
   is not given, current mode is returned.

   Mode "standard" is compatible with old :mod:`turtle`.  Mode "logo" is
   compatible with most Logo turtle graphics.  Mode "world" uses user-defined
   "world coordinates". **Attention**: in this mode angles appear distorted if
   ``x/y`` unit-ratio doesn't equal 1.

   ============ ========================= ===================
       Mode      Initial turtle heading     positive angles
   ============ ========================= ===================
    "standard"    to the right (east)       counterclockwise
      "logo"        upward    (north)         clockwise
   ============ ========================= ===================

   .. doctest::
      :skipif: _tkinter is None

      >>> mode("logo")   # resets turtle heading to north
      >>> mode()
      'logo'
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 1940)

Unknown directive type "function".

```
.. function:: colormode(cmode=None)

   :param cmode: one of the values 1.0 or 255

   Return the colormode or set it to 1.0 or 255.  Subsequently *r*, *g*, *b*
   values of color triples have to be in the range 0..*cmode*.

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.colormode(1)
      >>> turtle.pencolor(240, 160, 80)
      Traceback (most recent call last):
           ...
      TurtleGraphicsError: bad color sequence: (240, 160, 80)
      >>> screen.colormode()
      1.0
      >>> screen.colormode(255)
      >>> screen.colormode()
      255
      >>> turtle.pencolor(240,160,80)
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1963)

Unknown directive type "function".

```
.. function:: getcanvas()

   Return the Canvas of this TurtleScreen.  Useful for insiders who know what to
   do with a Tkinter Canvas.

   .. doctest::
      :skipif: _tkinter is None

      >>> cv = screen.getcanvas()
      >>> cv
      <turtle.ScrolledCanvas object ...>
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1976)

Unknown directive type "function".

```
.. function:: getshapes()

   Return a list of names of all currently available turtle shapes.

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.getshapes()
      ['arrow', 'blank', 'circle', ..., 'turtle']
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 1987)

Unknown directive type "function".

```
.. function:: register_shape(name, shape=None)
              addshape(name, shape=None)

   There are three different ways to call this function:

   (1) *name* is the name of a gif-file and *shape* is ``None``: Install the
       corresponding image shape. ::

       >>> screen.register_shape("turtle.gif")

       .. note::
          Image shapes *do not* rotate when turning the turtle, so they do not
          display the heading of the turtle!

   (2) *name* is an arbitrary string and *shape* is a tuple of pairs of
       coordinates: Install the corresponding polygon shape.

       .. doctest::
          :skipif: _tkinter is None

          >>> screen.register_shape("triangle", ((5,-3), (0,5), (-5,-3)))

   (3) *name* is an arbitrary string and *shape* is a (compound) :class:`Shape`
       object: Install the corresponding compound shape.

   Add a turtle shape to TurtleScreen's shapelist.  Only thusly registered
   shapes can be used by issuing the command ``shape(shapename)``.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2016)

Unknown directive type "function".

```
.. function:: turtles()

   Return the list of turtles on the screen.

   .. doctest::
      :skipif: _tkinter is None

      >>> for turtle in screen.turtles():
      ...     turtle.color("red")
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2027)

Unknown directive type "function".

```
.. function:: window_height()

   Return the height of the turtle window. ::

       >>> screen.window_height()
       480
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2035)

Unknown directive type "function".

```
.. function:: window_width()

   Return the width of the turtle window. ::

       >>> screen.window_width()
       640
```

Methods specific to Screen, not inherited from TurtleScreen

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2048)

Unknown directive type "function".

```
.. function:: bye()

   Shut the turtlegraphics window.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2053)

Unknown directive type "function".

```
.. function:: exitonclick()

   Bind ``bye()`` method to mouse clicks on the Screen.


   If the value "using_IDLE" in the configuration dictionary is ``False``
   (default value), also enter mainloop.  Remark: If IDLE with the ``-n`` switch
   (no subprocess) is used, this value should be set to ``True`` in
   :file:`turtle.cfg`.  In this case IDLE's own mainloop is active also for the
   client script.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2065)

Unknown directive type "function".

```
.. function:: setup(width=_CFG["width"], height=_CFG["height"], startx=_CFG["leftright"], starty=_CFG["top

   Set the size and position of the main window.  Default values of arguments
   are stored in the configuration dictionary and can be changed via a
   :file:`turtle.cfg` file.

   :param width: if an integer, a size in pixels, if a float, a fraction of the
                 screen; default is 50% of screen
   :param height: if an integer, the height in pixels, if a float, a fraction of
                  the screen; default is 75% of screen
   :param startx: if positive, starting position in pixels from the left
                  edge of the screen, if negative from the right edge, if ``None``,
                  center window horizontally
   :param starty: if positive, starting position in pixels from the top
                  edge of the screen, if negative from the bottom edge, if ``None``,
                  center window vertically

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.setup (width=200, height=200, startx=0, starty=0)
      >>>              # sets window to 200x200 pixels, in upper left of screen
      >>> screen.setup(width=.75, height=0.5, startx=None, starty=None)
      >>>              # sets window to 75% of screen by 50% of screen and centers
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2091)

Unknown directive type "function".

```
.. function:: title(titlestring)

   :param titlestring: a string that is shown in the titlebar of the turtle
                       graphics window

   Set title of turtle window to *titlestring*.

   .. doctest::
      :skipif: _tkinter is None

      >>> screen.title("Welcome to the turtle zoo!")
```

Public classes

param canvas: a :class:`tkinter.Canvas`, a :class:`ScrolledCanvas` or a :class:`TurtleScreen`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2111); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2111); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2111); *backlink*

Unknown interpreted text role "class".

Create a turtle. The turtle has all methods described above as "methods of Turtle/RawTurtle".

Subclass of RawTurtle, has the same interface but draws on a default :class:`Screen` object created automatically when needed for the first time.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2120); *backlink*

Unknown interpreted text role "class".

param cv: a :class:`tkinter.Canvas`

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2126); *backlink*

Unknown interpreted text role "class".

Provides screen oriented methods like :func:`setbg` etc. that are described above.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2128); *backlink*

Unknown interpreted text role "func".

Subclass of TurtleScreen, with :ref:`four methods added <screenspecific>`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2133); *backlink*

Unknown interpreted text role "ref".

param master: some Tkinter widget to contain the ScrolledCanvas, i.e. a Tkinter-canvas with scrollbars added

Used by class Screen, which thus automatically provides a ScrolledCanvas as playground for the turtles.

param type_: one of the strings "polygon", "image", "compound"

Data structure modeling shapes. The pair `(type_, data)` must follow this specification:

*type_*	*data*
"polygon"	a polygon-tuple, i.e. a tuple of pairs of coordinates
"image"	an image (in this form only used internally!)
"compound"	`None` (a compound shape has to be constructed using the :meth:`addcomponent` method) **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)turtle.rst`, line 2158); *backlink*** Unknown interpreted text role "meth".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)turtle.rst`, line 2161)

Unknown directive type "method".

```
.. method:: addcomponent(poly, fill, outline=None)

   :param poly: a polygon, i.e. a tuple of pairs of numbers
   :param fill: a color the *poly* will be filled with
   :param outline: a color for the poly's outline (if given)

   Example:

   .. doctest::
      :skipif: _tkinter is None

      >>> poly = ((0,0),(10,-5),(0,10),(-10,-5))
      >>> s = Shape("compound")
      >>> s.addcomponent(poly, "red", "blue")
      >>> # ... add more components and then use register_shape()

   See :ref:`compoundshapes`.
```

A two-dimensional vector class, used as a helper class for implementing turtle graphics. May be useful for turtle graphics programs too. Derived from tuple, so a vector is a tuple!

Provides (for *a*, *b* vectors, *k* number):

- `a + b` vector addition
- `a - b` vector subtraction
- `a * b` inner product
- `k * a` and `a * k` multiplication with scalar
- `abs(a)` absolute value of a
- `a.rotate(angle)` rotation

Help and configuration

How to use help

The public methods of the Screen and Turtle classes are documented extensively via docstrings. So these can be used as online-help via the Python help facilities:

- When using IDLE, tooltips show the signatures and first lines of the docstrings of typed in function-/method calls.
- Calling :func:`help` on methods or functions displays the docstrings:

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)turtle.rst`, line 2209); *backlink*

  Unknown interpreted text role "func".

```
>>> help(Screen.bgcolor)
Help on method bgcolor in module turtle:

bgcolor(self, *args) unbound turtle.Screen method
    Set or return backgroundcolor of the TurtleScreen.

    Arguments (if given): a color string or three numbers
    in the range 0..colormode or a 3-tuple of such numbers.


      >>> screen.bgcolor("orange")
      >>> screen.bgcolor()
      "orange"
      >>> screen.bgcolor(0.5,0,0.5)
      >>> screen.bgcolor()
      "#800080"

>>> help(Turtle.penup)
Help on method penup in module turtle:

penup(self) unbound turtle.Turtle method
    Pull the pen up -- no drawing when moving.
    Aliases: penup | pu | up

    No argument

    >>> turtle.penup()
```

- The docstrings of the functions which are derived from methods have a modified form:

```
>>> help(bgcolor)
Help on function bgcolor in module turtle:

bgcolor(*args)
    Set or return backgroundcolor of the TurtleScreen.

    Arguments (if given): a color string or three numbers
    in the range 0..colormode or a 3-tuple of such numbers.

    Example::

      >>> bgcolor("orange")
      >>> bgcolor()
      "orange"
      >>> bgcolor(0.5,0,0.5)
      >>> bgcolor()
      "#800080"

>>> help(penup)
Help on function penup in module turtle:

penup()
    Pull the pen up -- no drawing when moving.

    Aliases: penup | pu | up

    No argument

    Example:
    >>> penup()
```

These modified docstrings are created automatically together with the function definitions that are derived from the methods at import time.

Translation of docstrings into different languages

There is a utility to create a dictionary the keys of which are the method names and the values of which are the docstrings of the public methods of the classes Screen and Turtle.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2285)

Unknown directive type "function".

```
.. function:: write_docstringdict(filename="turtle_docstringdict")

   :param filename: a string, used as filename

   Create and write docstring-dictionary to a Python script with the given
   filename.  This function has to be called explicitly (it is not used by the
   turtle graphics classes).  The docstring dictionary will be written to the
   Python script :file:`{filename}.py`.  It is intended to serve as a template
   for translation of the docstrings into different languages.
```

If you (or your students) want to use :mod:`turtle` with online help in your native language, you have to translate the docstrings and save the resulting file as e.g. :file:`turtle_docstringdict_german.py`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2295); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2295); *backlink*

Unknown interpreted text role "file".

If you have an appropriate entry in your :file:`turtle.cfg` file this dictionary will be read in at import time and will replace the original English docstrings.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2299); *backlink*

Unknown interpreted text role "file".

At the time of this writing there are docstring dictionaries in German and in Italian. (Requests please to glingl@aon.at.)

How to configure Screen and Turtles

The built-in default configuration mimics the appearance and behaviour of the old turtle module in order to retain best possible compatibility with it.

If you want to use a different configuration which better reflects the features of this module or which better fits to your needs, e.g. for use in a classroom, you can prepare a configuration file `turtle.cfg` which will be read at import time and modify the configuration according to its settings.

The built in configuration would correspond to the following turtle.cfg:

```
width = 0.5
height = 0.75
leftright = None
topbottom = None
canvwidth = 400
canvheight = 300
mode = standard
colormode = 1.0
delay = 10
undobuffersize = 1000
shape = classic
pencolor = black
fillcolor = black
resizemode = noresize
visible = True
language = english
exampleturtle = turtle
examplescreen = screen
title = Python Turtle Graphics
using_IDLE = False
```

Short explanation of selected entries:

- The first four lines correspond to the arguments of the :meth:`Screen.setup` method.

  > **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2343); *backlink***
  >
  > Unknown interpreted text role "meth".

- Line 5 and 6 correspond to the arguments of the method :meth:`Screen.screensize`.

  > **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2345); *backlink***
  >
  > Unknown interpreted text role "meth".

- *shape* can be any of the built-in shapes, e.g: arrow, turtle, etc. For more info try `help(shape)`.
- If you want to use no fillcolor (i.e. make the turtle transparent), you have to write `fillcolor = ""` (but all nonempty strings must not have quotes in the cfg-file).
- If you want to reflect the turtle its state, you have to use `resizemode = auto`.
- If you set e.g. `language = italian` the docstringdict :file:`turtle_docstringdict_italian.py` will be loaded at import time (if present on the import path, e.g. in the same directory as :mod:`turtle`).

  > **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2354); *backlink***
  >
  > Unknown interpreted text role "file".

  > **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2354); *backlink***
  >
  > Unknown interpreted text role "mod".

- The entries *exampleturtle* and *examplescreen* define the names of these objects as they occur in the docstrings. The transformation of method-docstrings to function-docstrings will delete these names from the docstrings.
- *using_IDLE*: Set this to `True` if you regularly work with IDLE and its `-n` switch ("no subprocess"). This will prevent :func:`exitonclick` to enter the mainloop.

  > **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2361); *backlink***
  >
  > Unknown interpreted text role "func".

There can be a :file:`turtle.cfg` file in the directory where :mod:`turtle` is stored and an additional one in the current working directory. The latter will override the settings of the first one.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2365); *backlink*

Unknown interpreted text role "file".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2365); *backlink*

Unknown interpreted text role "mod".

The :file:`Lib/turtledemo` directory contains a :file:`turtle.cfg` file. You can study it as an example and see its effects when running the demos (preferably not from within the demo-viewer).

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2369); *backlink*

Unknown interpreted text role "file".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2369); *backlink*

Unknown interpreted text role "file".

:mod:`turtledemo` --- Demo scripts

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2374); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2377)

Unknown directive type "module".

```
.. module:: turtledemo
   :synopsis: A viewer for example turtle scripts
```

The :mod:`turtledemo` package includes a set of demo scripts. These scripts can be run and viewed using the supplied demo viewer as follows:

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2380); *backlink*

Unknown interpreted text role "mod".

```
python -m turtledemo
```

Alternatively, you can run the demo scripts individually. For example,

```
python -m turtledemo.bytedesign
```

The :mod:`turtledemo` package directory contains:

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2389); *backlink*

Unknown interpreted text role "mod".

- A demo viewer :file:`__main__.py` which can be used to view the sourcecode of the scripts and run them at the same time.

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2391); *backlink*

  Unknown interpreted text role "file".

- Multiple scripts demonstrating different features of the :mod:`turtle` module. Examples can be accessed via the Examples menu. They can also be run standalone.

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2393); *backlink*

  Unknown interpreted text role "mod".

- A :file:`turtle.cfg` file which serves as an example of how to write and use such files.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2396); *backlink*

Unknown interpreted text role "file".

The demo scripts are:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2401)

Unknown directive type "tabularcolumns".

```
.. tabularcolumns:: |l|L|L|
```

Name	Description	Features
bytedesign	complex classical turtle graphics pattern	:func:`tracer`, delay, :func:`update` **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2407); *backlink*** Unknown interpreted text role "func". **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2407); *backlink*** Unknown interpreted text role "func".
chaos	graphs Verhulst dynamics, shows that computer's computations can generate results sometimes against the common sense expectations	world coordinates
clock	analog clock showing time of your computer	turtles as clock's hands, ontimer
colormixer	experiment with r, g, b	:func:`ondrag` **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2419); *backlink*** Unknown interpreted text role "func".
forest	3 breadth-first trees	randomization
fractalcurves	Hilbert & Koch curves	recursion
lindenmayer	ethnomathematics (indian kolams)	L-System
minimal_hanoi	Towers of Hanoi	Rectangular Turtles as Hanoi discs (shape, shapesize)
nim	play the classical nim game with three heaps of sticks against the computer.	turtles as nimsticks, event driven (mouse, keyboard)

Name	Description	Features
paint	super minimalistic drawing program	:func:`onclick` **System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2436); ***backlink*** Unknown interpreted text role "func".
peace	elementary	turtle: appearance and animation
penrose	aperiodic tiling with kites and darts	:func:`stamp` **System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2442); ***backlink*** Unknown interpreted text role "func".
planet_and_moon	simulation of gravitational system	compound shapes, :class:`Vec2D` **System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2445); ***backlink*** Unknown interpreted text role "class".
round_dance	dancing turtles rotating pairwise in opposite direction	compound shapes, clone shapesize, tilt, get_shapepoly, update
sorting_animate	visual demonstration of different sorting methods	simple alignment, randomization
tree	a (graphical) breadth first tree (using generators)	:func:`clone` **System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2455); ***backlink*** Unknown interpreted text role "func".
two_canvases	simple design	turtles on two canvases

Name	Description	Features
wikipedia	a pattern from the wikipedia article on turtle graphics	:func:`clone`, :func:`undo` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2461); *backlink* Unknown interpreted text role "func". System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2461); *backlink* Unknown interpreted text role "func".
yinyang	another elementary example	:func:`circle` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst, line 2464); *backlink* Unknown interpreted text role "func".

Have fun!

Changes since Python 2.6

- The methods :meth:`Turtle.tracer`, :meth:`Turtle.window_width` and :meth:`Turtle.window_height` have been eliminated. Methods with these names and functionality are now available only as methods of :class:`Screen`. The functions derived from these remain available. (In fact already in Python 2.6 these methods were merely duplications of the corresponding :class:`TurtleScreen`/:class:`Screen`-methods.)

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2472); *backlink*

  Unknown interpreted text role "meth".

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2472); *backlink*

  Unknown interpreted text role "meth".

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2472); *backlink*

  Unknown interpreted text role "meth".

  System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-

resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2472); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2472); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2472); *backlink*

Unknown interpreted text role "class".

- The method :meth:`Turtle.fill` has been eliminated. The behaviour of :meth:`begin_fill` and :meth:`end_fill` have changed slightly: now every filling-process must be completed with an `end_fill()` call.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2480); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2480); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2480); *backlink*

Unknown interpreted text role "meth".

- A method :meth:`Turtle.filling` has been added. It returns a boolean value: `True` if a filling process is under way, `False` otherwise. This behaviour corresponds to a `fill()` call without arguments in Python 2.6.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2485); *backlink*

Unknown interpreted text role "meth".

Changes since Python 3.0

- The methods :meth:`Turtle.shearfactor`, :meth:`Turtle.shapetransform` and :meth:`Turtle.get_shapepoly` have been added. Thus the full range of regular linear transforms is now available for transforming turtle shapes. :meth:`Turtle.tiltangle` has been enhanced in functionality: it now can be used to get or set the tiltangle. :meth:`Turtle.settiltangle` has been deprecated.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2493); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2493); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2493); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst, line 2493); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2493); *backlink*

Unknown interpreted text role "meth".

- The method :meth:`Screen.onkeypress` has been added as a complement to :meth:`Screen.onkey` which in fact binds actions to the keyrelease event. Accordingly the latter has got an alias: :meth:`Screen.onkeyrelease`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2500); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2500); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2500); *backlink*

Unknown interpreted text role "meth".

- The method :meth:`Screen.mainloop` has been added. So when working only with Screen and Turtle objects one must not additionally import :func:`mainloop` anymore.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2504); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2504); *backlink*

Unknown interpreted text role "func".

- Two input methods has been added :meth:`Screen.textinput` and :meth:`Screen.numinput`. These popup input dialogs and return strings and numbers respectively.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2508); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2508); *backlink*

Unknown interpreted text role "meth".

- Two example scripts :file:`tdemo_nim.py` and :file:`tdemo_round_dance.py` have been added to the :file:`Lib/turtledemo` directory.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2512); *backlink*

Unknown interpreted text role "file".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2512); *backlink*

Unknown interpreted text role "file".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)turtle.rst`, line 2512); *backlink*

Unknown interpreted text role "file".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library) turtle.rst`, line 2516)

Unknown directive type "doctest".

```
.. doctest::
   :skipif: _tkinter is None
   :hide:

   >>> for turtle in turtles():
   ...      turtle.reset()
   >>> turtle.penup()
   >>> turtle.goto(-200,25)
   >>> turtle.pendown()
   >>> turtle.write("No one expects the Spanish Inquisition!",
   ...      font=("Arial", 20, "normal"))
   >>> turtle.penup()
   >>> turtle.goto(-100,-50)
   >>> turtle.pendown()
   >>> turtle.write("Our two chief Turtles are...",
   ...      font=("Arial", 16, "normal"))
   >>> turtle.penup()
   >>> turtle.goto(-450,-75)
   >>> turtle.write(str(turtles()))
```