Overview

In this lesson, you will learn how to draw **lines directly on the Tk canvas** using the turtle library. Instead of moving a turtle, you will create lines by specifying exact points. You will also learn how to control the **thickness**, **color**, and several **special line settings**.

Important Information

When you draw directly on the canvas, you are no longer giving movement commands. Instead, you tell the canvas exactly **where** to draw.

Creating a Line

Lines are drawn using:

```
canvas.create_line(x1, y1, x2, y2)
```

- `(x1, y1)` is the starting point
- `(x2, y2)` is the ending point

The coordinates are in **screen pixels**:

- `(0, 0)` is the center of the canvas
- X increases to the right
- Y increases downward

Changing Line Thickness

You can change how thick a line is using the `width` setting:

```
canvas.create_line(x1, y1, x2, y2, width=5)
```

- Larger values make thicker lines
- Smaller values make thinner lines

Changing Line Color

You can change the color of a line using the `fill` setting:

```
canvas.create_line(x1, y1, x2, y2, fill="red")
```

Common color names include:

- `"black"`
- `"red"`
- `"blue"`
- `"green"`
- `"purple"`

Dash Lines

Dashed lines can be created using the `dash` setting:

```
canvas.create_line(x1, y1, x2, y2, dash=(10, 5))
```

- The first number is the length of the dash
- The second number is the space between dashes

NOTE: On some systems the dashes have a limited number of options. Likely your dashes will be preconfigured and the numbers will not behave as expected. Instead, the numbers you input will try to use a predefined dash pattern.

Arrows

You can add arrows to lines using the `arrow` setting:

```
canvas.create_line(x1, y1, x2, y2, arrow="last")
```

Arrow options:

- `"first"` → arrow at the start
- `"last"` → arrow at the end
- `"both"` → arrows at both ends

Arrowhead Size

You can control the arrowhead shape using `arrowshape`:

```
canvas.create_line(x1, y1, x2, y2, arrow="last", arrowshape=(16, 20, 6))
```

The three numbers represent:

1. Arrowhead length
2. Arrowhead width
3. Arrow shaft width

Smooth Lines

You can smooth a line using the `smooth` setting:

```
canvas.create_line(x1, y1, x2, y2, smooth=True)
```

This is most useful when a line has multiple points, which will be explored in later lessons.

Set Up

Create a new Python file called `canvas_lines.py`.

Copy, Change, Challenge

Copy

Copy and run the following code.

```
import turtle

screen = turtle.Screen()
canvas = screen.getcanvas()

canvas.create_line(50, 50, 250, 50)

turtle.done()
```

You should see a simple horizontal line drawn on the canvas.

Change

Modify the line so that:

- The line is thicker
- The line is a different color
- The line is dashed

Run the program again and observe how each setting affects the line.

Challenge

Add additional lines so that:

- One line has an arrow
- One line uses a custom arrowhead size
- One line uses the `smooth` setting