CompSci 101 Lab 9
tkinter
and nested for loops

Drawing Patterns using nested for loops

- A nested loop is a loop inside a loop.
- The "inner loop" sits inside the "outer loop"
- Nested loops are often used when working with two dimensions, e.g. printing patterns

Drawing a simple pattern

```
for row in range(3):                    # Outer Loop
    for column in range(10):              # Inner Loop
        print("x", end="")
    print()
```

```
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
```

Another simple pattern

```
for row in range(3):
    print("Row:" + str(row), end=" ")
    for column in range(10):
        print(column,end="")
    print()
```

```
Row:0 0123456789
Row:1 0123456789
Row:2 0123456789
```

Another simple pattern

```
for row in range(3):
    for column in range(10):
        if column % 2 == 0:
            print("e", end="")
        else:
            print("o", end="")
    print()
```

```
eoeoeoeoeo
eoeoeoeoeo
eoeoeoeoeo
```

GUI programming using tkinter

- Tk is a platform independent windowing toolkit

- Available to us through the tkinter package.

- Tk provides the definitions of many “widgets” (labels, buttons, text boxes, etc.)

Canvas Widget

- The Canvas widget is the only widget that we will be using in this lab and also in the assignment.

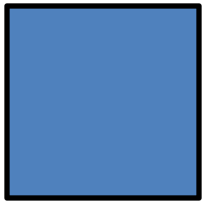
- The canvas is a general purpose widget
  - used to display graphs and other drawings.

- Each pixel in the Canvas area has an x position (across the canvas) and a y position (down the canvas).

- Position (0, 0) is the top left corner of the canvas.

Creating a rectangle

- **create_rectangle**(x0, y0, x1, y1, *options*)
- The rectangle is defined by 2 points:
  - (x0, y0) the top left position
  - (x1, y1) the bottom right position.

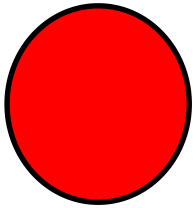


Example:

```
canvas.create_rectangle(50, 100, 150, 200,
                        fill = "blue", outline ="black")
```

Creating an oval

- **create_oval**(x0, y0, x1, y1, *options*)
- The oval is defined by 2 points:
  - (x0, y0) the top left position of the bounding rectangle
  - (x1, y1) the bottom right position of the bounding rectangle.

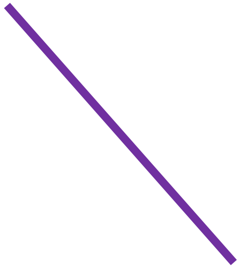


Example:

```
canvas.create_oval(50, 100, 150, 200,
                      fill = “red", outline = "black")
```

Creating a line

- **create_line**(x0, y0, x1, y1, *options*)
- The line is defined by 2 points
  - (x0, y0) and (x1, y1).

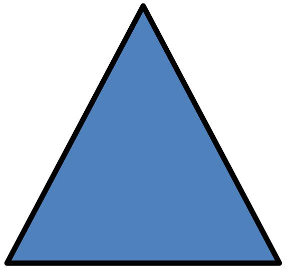


Example:

```
canvas.create_line(100, 250, 150, 275,
                          fill = "purple", width = 3)
```

Creating a polygon

- **create_polygon**(coordinates of points, *options*)
- The polygon is defined by a series of points:
  - (x0, y0, x1, y1, …. xn, yn).



Example (creating a triangle):

```
canvas.create_polygon(50,100, 125,25,200,100,
                            fill = "blue", outline = "black")
```

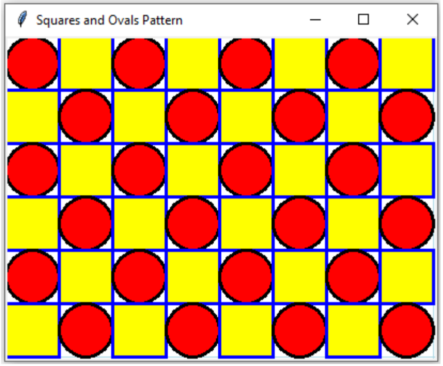
Creating text

- **create_text**(coordinates of position, *options)*
  - Draws text in the canvas.

Example:

```
my_font = ("Courier", 12, "bold")
canvas.create_text(50, 100, text = "Hello",
                 fill = "green", font = my_font)
```

Pattern in Canvas Widget



Squares and Ovals Pattern

```
def draw_pattern_in_canvas(canvas, size):
    top = 0
    for row in range(6):
        left = 0
        draw_oval = row % 2 == 0
        for column in range(8):
            if draw_oval == True:
                canvas.create_oval(left, top, left + size, top + size,
                                   fill = "red", outline = "black", width = 3)
            else:
                canvas.create_rectangle(left, top, left + size, top + size,
                                   fill = "yellow", outline = "blue", width = 3)
            draw_oval = not draw_oval
            left += size
        top += size
```

Next Week

There is no lab next week.

This is the last lab for this semester.

Good luck for all your final assessments ☺