SAMS Programming Workshop - Python Cheat Sheet

Basic Operations

```
1 + 2     # addition
2 - 1     # subtraction
2 * 3     # multiplication
4 / 2     # division
2 ** 3    # raise to power
"a" + "b" # combines two strings
```

Comparison Operations

```
2 < 3      # less than
"a" > "b"  # greater than
"a" <= "b" # less than or equal to
2 >= 3     # greater than or equal to
"a" == "b" # is equal to
2 != 3     # is not equal to
```

Boolean Operations

```
True and False # both must be True
True or False  # at least one True
not True       # flips boolean value
```

Input and Output

```
# prints to the console
print("Hello World")

# prints multiple items
print("a", "b", "c")

# asks the user for input in console
s = input("Enter a thing: ")
```

Built-in Functions

```
x = int(s) # casts a string to an int
x = float(s) # casts a string to a float
s = str(x) # casts an int to a string
x = len(s) # finds num of letters in str
x = max(a, b, c) # max of given numbers
x = min(a, b, c) # min of given numbers
x = round(y, d) # rounds y to d sig-digs
```

Variables

```
# assigns var x to hold the value 5
x = 5

# uses the value in var x
print(x - 2)
```

Functions

```
# sets up a new function, name
# it takes params p1, p2, ... as input
# it returns returnVal as output
def name(p1, p2, ...):
    # do stuff here
    return returnVal

# call a function name on vals a, b
result = name(a, b)
```

Conditionals

```
# only runs code in the block
# if the expression is True
if option == True:
    print("here")

# run code in a single branch
# based on the boolean expressions
if option1 == True:
    print("branch 1")
elif option2 == True:
    print("branch 2")
else:
    print("branch 3")
```

Loops

```
# loops until the test is False
i = start
while i < end:
    print(i)
    i = i + step

# breaks out of loop at some input
while True:
    val = input("Enter: ")
    if val == "something":
        break
    print(val)

# loops over the given range
# with start, end, step
for i in range(start, end, step):
    print(i)

# loops over the chars in a string
for c in string:
    print(c)
```

SAMS Programming Workshop - Python Cheat Sheet

Tkinter Starter Code

```
# use this to create a window to draw graphics in
from tkinter import *
root = Tk()
width, height = 400, 400
canvas = Canvas(root, width=width, height=height)
canvas.configure(bd=0, highlightthickness=0)
canvas.pack()

# Put your code here!

root.mainloop()
```

Tkinter Graphics

```
# draws a rectangle between coords (left, top) and (right, bottom)
canvas.create_rectangle(left, top, right, bottom)

# draws an oval in the bounding box with coords (L, T) and (R, B)
canvas.create_oval(L, T, R, B)

# draws the given text centered at the given coordinate (x, y)
canvas.create_text(x, y, text="sample")

# draws a line between the given points (x1, y1) and (x2, y2)
canvas.create_line(x1, y1, x2, y2)

# draws a polygon by connecting the given points with lines
canvas.create_polygon(x1, y1, x2, y2, x3, y3, ...)
```

Tkinter Optional Parameters

```
# changes the color of the drawn shape
canvas.create_rectangle(left, top, right, bottom, fill="red")

# changes the outline color of the drawn shape
canvas.create_rectangle(left, top, right, bottom, outline="yellow")

# changes the pixel width of the drawn line or shape's border
canvas.create_line(x1, y1, x2, y2, width=5)

# changes the font of the drawn text- "font-name font-size font-style"
canvas.create_text(x, y, text="sample", font="Times 30 bold")

# changes the anchor point for the drawn text
canvas.create_text(x, y, text="sample", anchor=NW)
```