Chapter 5: Sequences Part 1: Strings and Lists

Jan 28, 2020

Today’s Outline

- Review:
  - Simple event-driven programming with graphics.py
  - Review for Quiz 3
- Strings
  - String indexing
  - String methods
- Lists

Lab 1 Marks Returned

Everyone did well on Lab 1. Good job!

If you have any questions about your grade, please email me and I can check your assignment.

If you have any trouble understanding the questions in future labs/quizzes, please don’t hesitate to let me know.

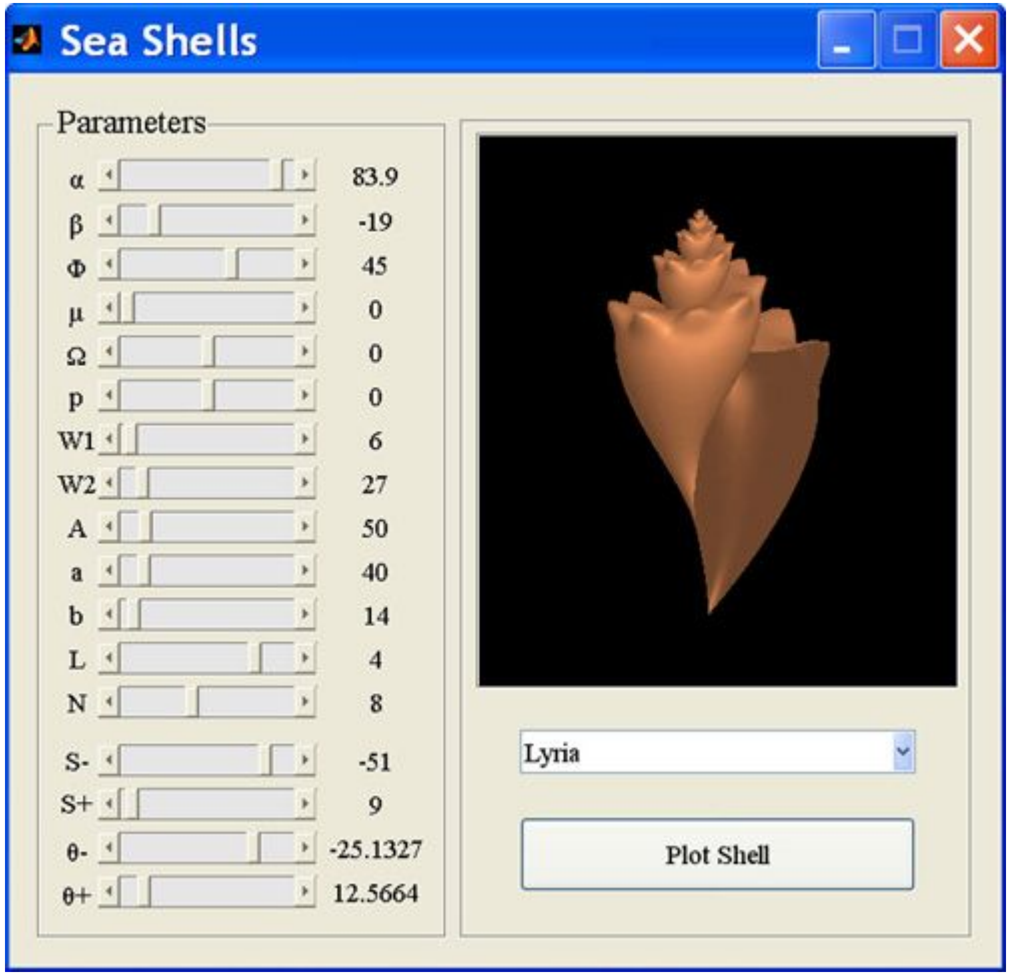
Interactive Graphics

event-driven programming:

The flow of the program is determined by user actions (ex. mouse clicks, keyboard entries, sensor inputs)

Ex: win.getMouse()

win.getKey()



Sea Shells

Parameters

α	83.9
β	-19
Φ	45
μ	0
Ω	0
p	0
W1	6
W2	27
A	50
a	40
b	14
L	4
N	8
S-	-51
S+	9
θ-	-25.1327
θ+	12.5664

Lyria

Mouse Click Inputs

Clicking the mouse, or pressing a key generates an **event** object.

For example: clicking a button generates a button event

The **getMouse()** method tells the window object to wait until the user clicks on the graphics window. The spot where the user clicked on the window is returned as a Point object.

Mouse Click Example

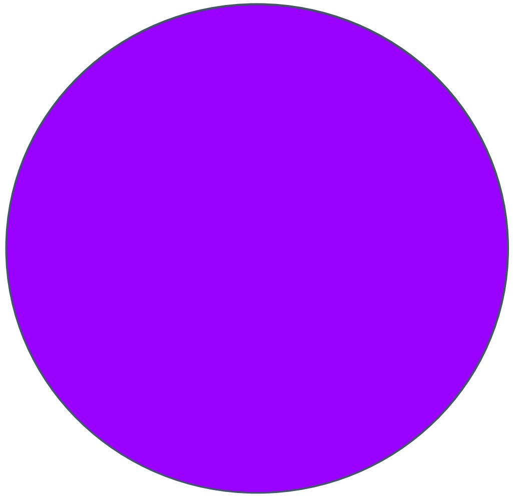
```
win = graphics.GraphWin ("Click Me!",250, 250)
for i in range (5):
    p = win.getMouse ( )
    print (" You clicked at:", p.getX(), p.getY())
```

GetKey() example

```
win = graphics.GraphWin ("Click and Type", 400, 400)
for i in range (10):
    p = win.getMouse()
    key = win.getKey()
    label = graphics.Text(p,key)
    label.draw (win)
```

Moving Ball Problem

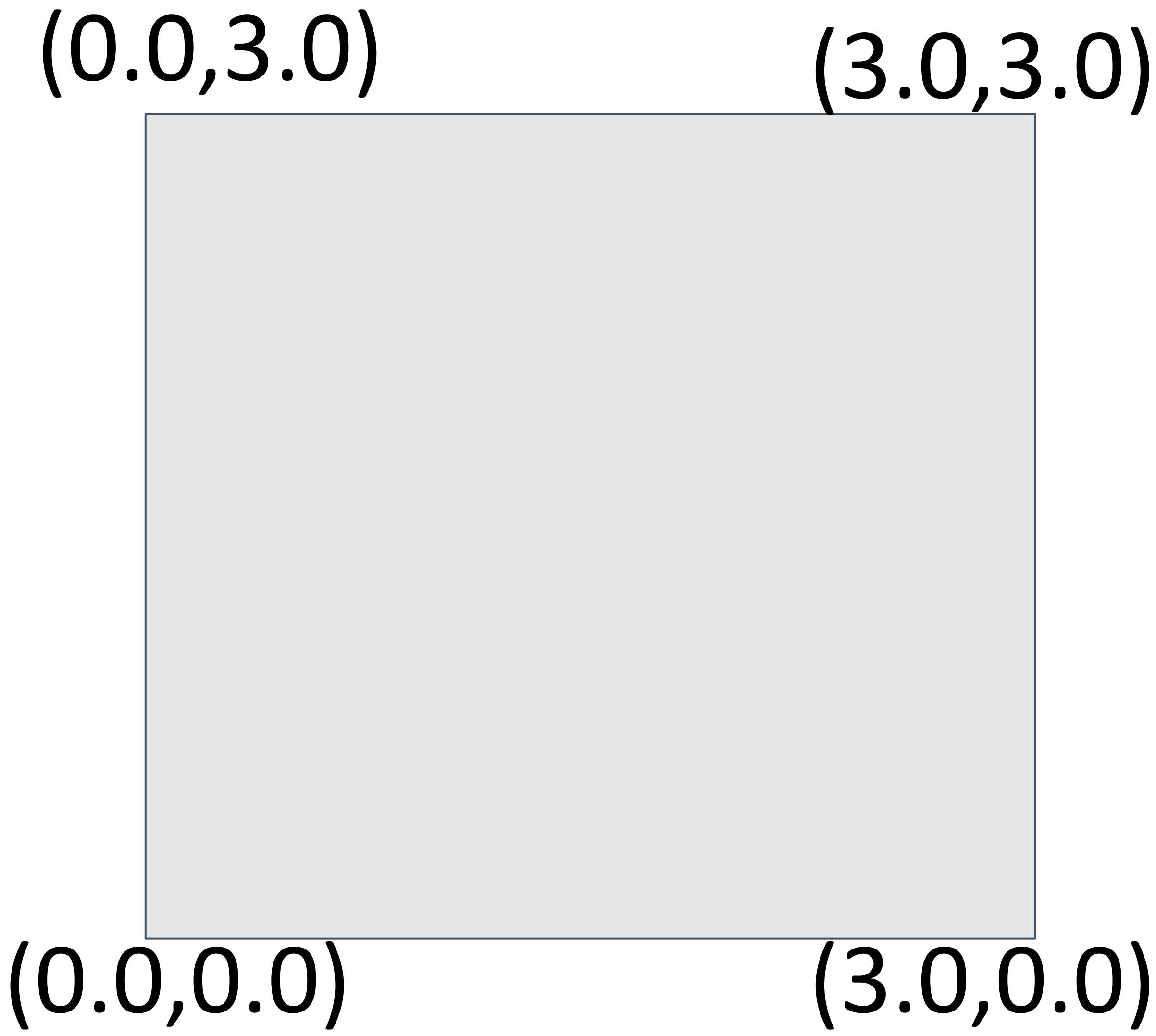
Develop a code that moves a purple ball to the location on the window where the mouse was clicked.



Coordinate Transformation Example

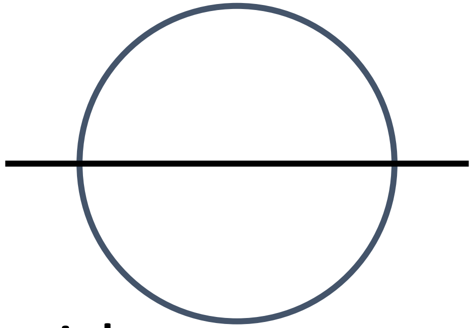
```
win = graphics.GraphWin()
win.setCoords(0.0, 0.0, 3.0 , 3.0 )
```

The .setCoords() method was used to set the coordinates of the lower left corner to (0.0, 0.0) and the upper right corner to (3.0, 3.0),



Intersection

$$x = \pm\sqrt{r^2 - y^2}$$



Write a program that computes the intersection of a circle with a horizontal line and displays the information textually and graphically.

Input: Radius of the circle and the y-intercept of the line.

Output: Draw a circle centered at (0, 0) with the given radius in a window with coordinates running from -10,-10 to 10, 10.

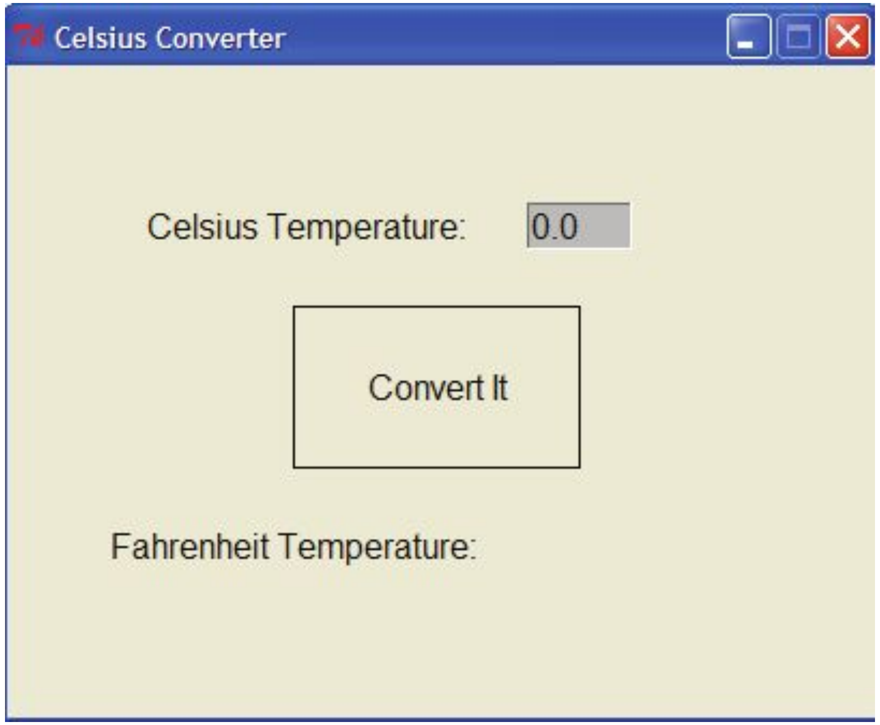
Draw a horizontal line across the window with the given y-intercept.

Draw the two points of intersection in red.

Print out the x values of the points of intersection.

Graphical User Interface (GUI)

Developing a graphical user interface (GUI) to convert a celsius value into fahrenheit.



Celsius Temperature: 0.0

Fahrenheit Temperature:

Quiz 3: Graphics

Quiz 3 will be due this Sunday, Feb 2

Quiz 3 Review

Which of the following best describes what an object is?

a. It is a geometric shape.

b. It is a group of methods.

c. It is a group of data.

d. It is a group of data along with methods for operating on that data.

Quiz 3 Review

If we import the graphics module using the statement **import graphics**, how do we create a Point object?

a. p = graphics.Point()

b. p = Point(2, 3)

c. p = create(Point(2, 3))

d. p = graphics.Point(2, 3)

Quiz 3 Review

If you have a Point object p and a GraphWin object win, how do you make the point p appear in the GraphWin window?

a. win.draw(p)

b. draw(p, win)

c. p.draw(win)

d. draw(win, p)

Quiz 3 Review

Which of the following classes have a draw method?

a. All of the classes in the graphics module.

b. The Point, Line, Circle, Oval, Rectangle, Text, and Entry classes.

c. Only the GraphWin class.

d. none of the above

Quiz 3 Review

How many Point objects does the following code create?

```
p1 = Point(2, 3)
p2 = Point(4, 5)
p3 = p1
p4 = p2
```

Select one:

a. 4 b. 3 c. 2 d. 1

Quiz 3 Review

What does the GraphWin getMouse() method return?

a. None

b. Two numeric values.

c. A GraphWin object.

d. A Point object.

Quiz 3 Review

If you have a GraphWin object win and execute win.setCoords(0, -100, 200, 100), where is the coordinate (0, 0)?

a. The middle of the left edge of the window.

b. The center of the window.

c. The bottom left of the window.

d. (0, 0) is not within the bounds of the window.

Quiz 3 Review

What color is color_rgb(255, 255, 0) ?

a) magenta (pink)
b) cyan (light blue)
c) yellow
d) orange

Quiz 3 Review

Which of the following graphics methods is a mutator?

a) getX()
b) GraphWin()
c) setFill()
d) getMouse()

Quiz 3 Review

What expression would create a line from (2,3) to (4,5)?

a) graphics.Line (2 , 3 , 4 , 5)

b) graphics.Line ((2 , 3) , (4 , 5))

c) graphics.Line (2 , 4 , 3 , 5)

d) graphics.Line (graphics.Point (2,3) , graphics.Point (4,5))

Sequences: Strings

Text in Python is stored as a **string** data type. A string is a **sequence** of characters.

```
greeting = "Hello World"
print(type(greeting))
```

Strings are **objects**.

Sequences: Strings

We already learned how to use strings:

```
firstName = input ( "Please enter your name : " )
print ( "Hello" , firstName)
```

String Indexing

H	e	l	l	o		W	o	r	l	d
0	1	2	3	4	5	6	7	8	9	10

The characters in a string can be accessed individually.

Note:

The string indexing begins at 0.

The space counts as a character in the string.

String Indexing

H	e	l	l	o		W	o	r	l	d
0	1	2	3	4	5	6	7	8	9	10

```
greeting = "Hello World"
print(greeting[0])
print(greeting[2])
print(greeting[5])
print(greeting[-1])
```

String Slices

H	e	l	l	o		W	o	r	l	d
0	1	2	3	4	5	6	7	8	9	10

```
greeting = "Hello World"

print(greeting[0:7])

print(greeting[3:5])

print(greeting[6:-1])
```

String Operations

```
superhero = "super" +"man"

greet3 = "hello"*3

L = len(greet3)
```

String for loop

```
for ch in "Hello World":
  print(ch, end = " ")
```

Username Generator

Develop a program that provides a user with a username.

The username pattern is the first letter of their first name, and their last name.

ex: Emma Farago

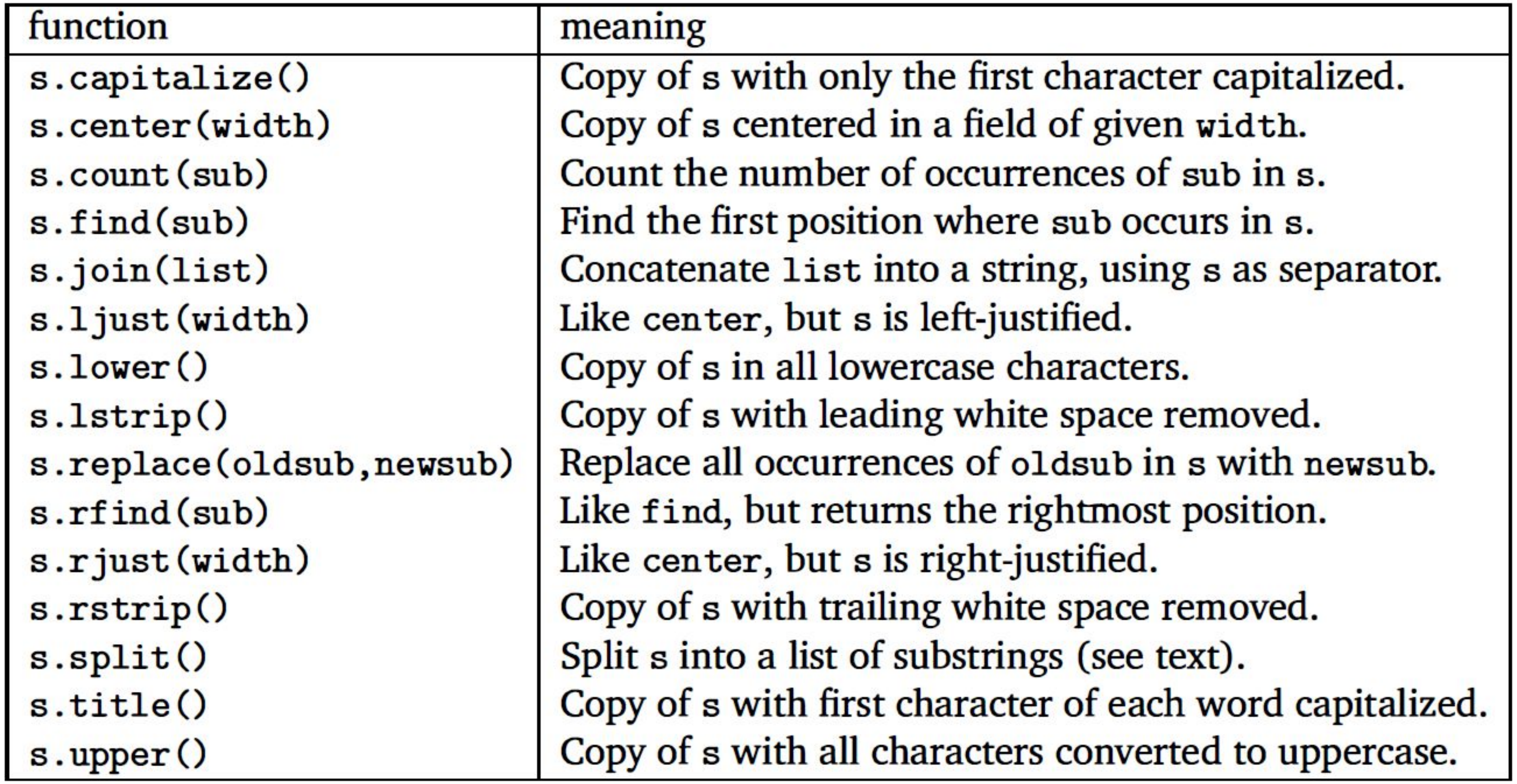
username: efarago

String Methods

Strings are **objects**.

We can therefore manipulate them using string methods.

String Methods



function	meaning
`s.capitalize()`	Copy of `s` with only the first character capitalized.
`s.center(width)`	Copy of `s` centered in a field of given `width`.
`s.count(sub)`	Count the number of occurrences of `sub` in `s`.
`s.find(sub)`	Find the first position where `sub` occurs in `s`.
`s.join(list)`	Concatenate `list` into a string, using `s` as separator.
`s.ljust(width)`	Like `center`, but `s` is left-justified.
`s.lower()`	Copy of `s` in all lowercase characters.
`s.lstrip()`	Copy of `s` with leading white space removed.
`s.replace(oldsub,newsub)`	Replace all occurrences of `oldsub` in `s` with `newsub`.
`s.rfind(sub)`	Like `find`, but returns the rightmost position.
`s.rjust(width)`	Like `center`, but `s` is right-justified.
`s.rstrip()`	Copy of `s` with trailing white space removed.
`s.split()`	Split `s` into a list of substrings (see text).
`s.title()`	Copy of `s` with first character of each word capitalized.
`s.upper()`	Copy of `s` with all characters converted to uppercase.

Table 5.2: Some string methods

Username Generator 2

Update the username generator program so that it works if the user types in their first name and last name as capitals.

Username Generator 2

Update the username generator program so that it works if the user types in their first name and last name as capitals.

Hint: use **s.lower()** method to make all letters in a string lowercase

String Slicing Month Example

Write a program that prints the abbreviation of a month given an INT from 1-12 that corresponds to the month.

1 = Jan

2 = Feb

and so on...

JANUARY

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

FEBRUARY

Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

MARCH

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

APRIL

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

MAY

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

JUNE

Su	Mo	Tu	We	Th	Fr	Sa
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

JULY

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

AUGUST

Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

SEPTEMBER

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

OCTOBER

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

NOVEMBER

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

DECEMBER

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

String Slicing Month Example

Write a program that prints the abbreviation of a month given an INT from 1-12 that corresponds to the month.

Instead of using decision structures, we can store all of the month abbreviations in a “lookup table”.

```
months = "JanFebMarAprMayJunJulAugSepOctNovDec"
```

String Slicing Month Example

```
months = "JanFebMarAprMayJunJulAugSepOctNovDec"
```

Jan: 1   Pos: 0

Feb: 2   Pos: 3

Mar: 3   Pos: 6

Lists

Strings are sequences of characters.

Lists are sequences of any type of object.

ex:

```
myList = [ 1 , "Spam" , 4.5 , "U"]
letterGrades = ["A", "B", "C", "D"]
marks = [99, 40, 74, 50]
```

List Example: Graphics objects

```
import graphics
win = graphics.GraphWin()
circ1 = graphics.Circle(graphics.Point(50,50),30)
circ2 = graphics.Circle(graphics.Point(100,100),30)
circ3 = graphics.Circle(graphics.Point(150,0),30)
circleList = [circ1,circ2,circ3]

for circles in circleList:
    circles.draw(win)
```

Lists vs. Strings

1) Lists can contain any type of objects. Strings are always sequences of characters.

2) Lists are mutable, but strings are not.

For example, the following code will not work:

```
myString = "hell0 world"
myString[4] = "o"
```

Month Example: Redo with a list

Write a program that prints the abbreviation of a month given an INT from 1-12 that corresponds to the month.

1 = Jan

2 = Feb

and so on...

JANUARY

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

FEBRUARY

Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

MARCH

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

APRIL

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

MAY

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

JUNE

Su	Mo	Tu	We	Th	Fr	Sa
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

JULY

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

AUGUST

Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

SEPTEMBER

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

OCTOBER

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

NOVEMBER

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

DECEMBER

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Month Example: Redo with a list

Write a program that prints the abbreviation of a month given an INT from 1-12 that corresponds to the month.

```
months = ["Jan", "Feb", "Mar", "Apr", "May",
          "Jun", "Jul","Aug","Sep","Oct",
          "Nov","Dec"]
```

Animal Example

Write a program that calculates the user’s Chinese zodiac animal based on their year of birth. There are 12 animals that repeat in a 12 year cycle. 2020 is the year of the rat.

```
animals = ["Rat", "Ox", "Tiger", "Rabbit", "Dragon",
        "Snake", "Horse", "Goat", "Monkey", "Rooster",
        "Dog","Pig"]
```