**Outline**

t.b.d.

**Objectives**

* tbd

**Materials**

* tbd

**Level 0: Teacher Demo of Sample Programs**

1. Sample program #1 is an example of a "Syntax Error". Follow the teacher demo and explain the characteristics of a syntax error. Consider the following criteria:  
   1. Did the program have an error before starting to run?
   2. Did the program encounter an error before it finished running?
   3. Did the program do what it was supposed to do?

1. Sample program #2 is an example of a "Run-time Error". Follow the teacher demo and explain the characteristics of a run-time error. Consider the following criteria:  
   1. Did the program have an error before starting to run?
   2. Did the program encounter an error before it finished running?
   3. Did the program do what it was supposed to do?

1. Sample program #3 is an example of a "Logic Error". Follow the teacher demo and explain the characteristics of a logic error. Consider the following criteria:  
   1. Did the program have an error before starting to run?
   2. Did the program encounter an error before it finished running?
   3. Did the program do what it was supposed to do?

**Level 1: Syntax Errors**

1. Research the definition of the word "Syntax". Summarize its meaning below and how it relates to computer languages and programming.

Syntax is referring to a programming language, syntax is a set of rules for grammar and spelling. Moreover, it means using character structures that a computer can interpret. For example, if a user tries to execute a command without proper syntax, it generates syntax error, usually causing the program to fail

1. Research the definition of a "Syntax Error" related to computer programming. Summarize this definition below.

Syntax error is an error in the syntax of a sequence of characters or tokens that is intended to be written in a particular programming language.

1. Explain why Sample Program #1 is an example of a "Syntax Error".

There is problem with the spelling/programming in the language.

1. Find and correct the syntax errors in Sample Program #1. Provide a listing of your corrected program below.
   * Use a "#" at the beginning of each line containing an error   
     to "Comment Out" the bad code
   * List the corrected code line underneath the commented out error line
2. import turtle
3. myPen = turtle.Turtle()
4. circleColors = [(196,196,0),(196,0,196),(0,196,196)]
5. def drawCircle(rgb) :
6. myPen.down( Missing bracet
7. myPen.color(rgb)
8. myPen.begin\_fill()
9. myPen.circle(8)
10. myPen.end\_fill()
11. myPen.up()
12. myPen.forward(22)
13. circleNumber = 0
14. for circleIndex in range(3) :
15. drawCircle(circleColours[circleNumber]) “circleColours”
16. circleNumber = circleNumber + 1

**Level 2: Run-time Errors**

1. Research the definition of a "Run-time Error" related to computer programming. Summarize this definition below.

A runtime error is a program error that occurs while the program is running. The term is often used in contrast to other types of program errors, such as syntax errors and compile time errors. There are many different types of runtime errors. One example is a logic error, which produces the wrong output.

1. Explain why Sample Program #2 is an example of a "Run-time Error".

Sample program 2 is a example of a run time error because the error occurred while the program was running. It stopped in the middle of the program.

1. Find and correct the run-time errors in Sample Program #2. Provide a listing of your corrected program below.
   * Use a "#" at the beginning of each line containing an error   
     to "Comment Out" the bad code
   * List the corrected code line underneath the commented out error line

import turtle

myPen = turtle.Turtle()

circleColours = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down()

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = 0 #made into 0 because the index starts from 0

for circleIndex in range(3) : #made it into 3 because the index only has 3 values

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

1. Explain the difference between a "syntax error" and a "run-time error".

The difference between a syntax error and a run-time error is a syntax error does not let the program run right away and the code has a spelling/programming language mistake. And a run-time error starts up the program but stops before completing the error which results in the program not doing what it was supposed to do.

**Level 3: Logic Errors**

1. Research the definition of a "Logic Error" related to computer programming. Summarize this definition below.

In computer programming, a logic error is a bug in a program that causes it to operate incorrectly, but not to terminate abnormally (or crash).

1. Explain why Sample Program #3 is an example of a "Logic Error".

Sample Program #3 is an example of a Logic error because although the program runs and python does not recognize any problem with the code.

1. Find and correct the logic errors in Sample Program #3. Provide a listing of your corrected program below.
   * Use a "#" at the beginning of each line containing an error   
     to "Comment Out" the bad code
   * List the corrected code line underneath the commented out error line   
     import turtle
   * myPen = turtle.Turtle()
   * circleColours = [(195,195,0),(195,0,195),(0,195,1965]
   * def drawCircle(rgb) :
   * myPen.down()
   * myPen.color(rgb) #missing code for color of circles
   * myPen.begin\_fill()
   * myPen.circle(8)
   * myPen.end\_fill()
   * myPen.up()
   * myPen.forward(22)
   * numCircles = 3
   * for circleIndex in range(3) : #changed to 3 because there are 3 circles
   * circleNumber = numCircles - circleIndex - 1
   * drawCircle(circleColours[circleNumber])
2. Explain the difference between a "logic error" and a "syntax error".

Syntax errors occur when a program does not conform to the grammar of a programming language, and the compiler cannot compile the source file. Logic error occur when a program does not do what the programmer expects it to do

1. Explain the difference between a "logic error" and a "run-time error".

Logic error occur when a program does not do what the programmer expects it to do. A run-time error starts the program but then stops it before it can finish. This means while the program was running, it found an error in the code which prevented the program from completing.

A run-time error starts the program but then stops it before it can finish

1. Explain the difference between a "logic error" and a "run-time error".

**Level 4: Your Sample Program**

1. Create a sample program to show the different types of programming errors. Provide your program listing below.
   * Your program must be of your own design and must be different from the sample programs provided in this module.
   * Your program must contain at least one example of each of: a syntax error, a run-time error, and a logic error.
   * Provide the corrected code in a comment underneath the error code (using a "#" at the beginning of the comment line).

import turtle

myPen = turtle.Turtle()

or i in range(4): # Syntax error, Fix: for i in range(4):

myPen.right(90)

myPen.forward(100)

myPen.left(90)

myPen.forward(50)

myPen.left(90)

myPen.forward(100)

myPen.up()

myPen.foward(250) # Run-time error, Fix; myPen.forward(250)

myPen.right(90)

myPen.down()

for x in range(3):

myPen.forward(100)

myPen.left(90)

myPen.forward(100)

myPen.up()

myPen.forward(600)

myPen.down()

myPen.color("blue")

myPen.circle(50)

myPen.color("re") #Logic Error, square supposed to turn out red, Fix: myPen.color("red")

myPen.forward(50)

myPen.left(90)

for h in range(4):

myPen.forward(100)

myPen.left(90)

**SAMPLE PROGRAM #1 - Syntax Error**

import turtle

myPen = turtle.Turtle()

circleColors = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down(

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = 0

for circleIndex in range(3) :

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

**SAMPLE PROGRAM #2 - Run-time Error**

import turtle

myPen = turtle.Turtle()

circleColours = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down()

myPen.color(rgb)

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = 1

for circleIndex in range(4) :

print(circleNumber)

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

**SAMPLE PROGRAM #3 - Logic Error**

import turtle

myPen = turtle.Turtle()

circleColours = [(196,196,0),(196,0,196),(0,196,196)]

def drawCircle(rgb) :

myPen.down()

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

numOfCircles = 2

for circleIndex in range(2) :

circleNumber = numOfCircles - circleIndex

print(circleNumber)

drawCircle(circleColours[circleNumber])