**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?  
      we got an error before program run
   2. Did the program encounter an error before it finished running?  
      it didn’t encounter because it didn’t work
   3. Did the program do what it was supposed to do?

No it didn’t because it did not work and didn’t do what it is supposed

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?  
      it started to run
   2. Did the program encounter an error before it finished running?  
      no it did not finished
   3. Did the program do what it was supposed to do?

No it didn’t do what it supposed because it was supposed to draw 3 circle

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?  
      no it draw2 black color circle
   2. Did the program encounter an error before it finished running?  
      no it didn’t encounter error only stop running
   3. Did the program do what it was supposed to do?

From the program view it did what is supposed to do but logically it did not do what 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.

A syntax error is a sequence of characters or tokens that is intended to be written in a particular **programming** language

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

**syntax** is an **error** in the **syntax** of arraignment of character or words is intended to be **written** in a particular language

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

Sample 2 is an syntax error because it didn’t run the program first of all so is an syntax error

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(
7. myPen.color(rgb)
8. #commentout
9. myPen.begin\_fill()
10. myPen.circle(8)
11. myPen.end\_fill()
12. myPen.up()
13. myPen.forward(22)
14. circleNumber = 0
15. for circleIndex in range(3) :
16. drawCircle(circleColours[circleNumber])
17. 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.

Is a program error that occurs while the program is running.

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

Sample 2 is a run time error because it stop encountering 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   
myPen = turtle.Turtle()

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

def drawCircle(rgb) :

myPen.down()

myPen.color(rgb)

#comment out

myPen.begin\_fill()

myPen.circle(8)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = 1

for circleIndex in range(4) :

#comment out

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

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

Inruntime the program will start to run but won’t complete finish it but in syntax error it won’t work at all because you probably messed up something.

**Level 3: Logic Errors**

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

A logic error is that in the program it did what it supposed to do but logically you messed up something and you didn’t notice

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

In sample program 3 it is an example of a logic error because there is no correction to be made but logically you have to because it just gives you to black circle

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 = [(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 = 3

for circleIndex in range(numOfCircles) :

   #circleNumber = numOfCircles - circleIndex - 1

   drawCircle(circleColours[circleIndex])

1. Explain the difference between a "logic error" and a "syntax error".

Difference between logic error is that it gives you the right answer but you did something wrong and the python doesn’t know and in syntax error is that it just doesn’t run

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

In logic error gives you right and in run time it start to stop but it won’t finish

**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).

Syntax error example#1

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) #didnt put in variable

myPen.end\_fill()

myPen.up()

myPen.forward(22)

circleNumber = ()

for circleIndex in range(3) :

drawCircle(circleColours[circleNumber])

circleNumber = circleNumber + 1

runtime error example#2

import turtle

r = turtle.Turtle()

r.pendown()

r.forward(50)

r.lt(45)

r.penup()

r.forward(n)

r.pendown()

r.forward(20)

Logic error example#3

import turtle

myPen = turtle.Turtle()

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

def drawCircle(rbg) :

myPen.down()

myPen.begin\_fill()

myPen.circle(15)

myPen.end\_fill()

myPen.up()

myPen.forward(22)

numOfCircles = 3

for circleIndex in range(numOfCircles) :

#circleNumber = numOfCircles - circleIndex - 1

drawCircle(circleColours[circleIndex])

**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) :

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 = 3

for circleIndex in range(2) :

circleNumber = numOfCircles - circleIndex - 1

drawCircle(circleColours[circleNumber])