**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?  
      Yes
   2. Did the program encounter an error before it finished running?  
      No
   3. Did the program do what it was supposed to do?

No

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?  
      No
   2. Did the program encounter an error before it finished running?  
      Yes
   3. Did the program do what it was supposed to do?

No

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
   2. Did the program encounter an error before it finished running?  
      No
   3. Did the program do what it was supposed to do?
      * No but there were no errors
      * Two black circles
      * No colorful

**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 means the arrangement of words and phrases to create well-formed sentences in a language.**

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

**Syntax error means a character or string incorrectly placed in a command or instructions that causes a failure in execution.**

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

**There is a mistake in the code because the code is not written correctly.**

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

**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( #bracket missing

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]) #spelt wrong

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.

**It 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".

**The error occurred while the program was running which caused the program to stop.**

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

**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 # Change it to 0

for circleIndex in range(4) :

drawCircle(circleColours[circleNumber])#Index is out of range

circleNumber = circleNumber + 1

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

**A program with a syntax error cannot be executed. A program with a runtime error can be executed under some conditions. Syntax errors are static detected by a compiler while runtime errors are dynamic and cannot be detected by a compiler.**

**Level 3: Logic Errors**

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

**A logic error is a bug in the program the causes it to operate incorrectly. It produced unintended output or other behavior although it may not immediately be recognized a such.**

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

**It is an error that python did not recognize.**

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(2) :

circleNumber = numOfCircles - circleIndex - 1

drawCircle(circleColours[circleNumber])

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

**Syntax is when something is written incorrectly in the code that the computer recognizes while logic is the same but the computer doesn’t recognize.**

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

**Run time error is an error that causes the program to stop running while logic is an error that the computer does not recognize.**

**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).

**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])