**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, the program did not run .
   2. Did the program encounter an error before it finished running?  
      Yes, it could not run.
   3. Did the program do what it was supposed to do?

No, it did not even run.

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, the program didn’t have any error. It started running and even made two of the three circles.
   2. Did the program encounter an error before it finished running?  
      Yes, the program did encounter an error.
   3. Did the program do what it was supposed to do?

No, the program didn’t do what it was supposed to do.

It only drew two circles while it was supposed to draw three.

It couldn’t finish because it encountered an error.

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, the program did not have an error before starting to run. It started running

smoothly.

* 1. Did the program encounter an error before it finished running?
* No, it did not encounter an error before it finished running.
  1. Did the program do what it was supposed to do?
* No, the program did not do what it was supposed to do.
* It only drew two circles which were black.

**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 the set of rules, principles, and processes that govern the structure of sentences in a given language, usually including word order.

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
* Syntax errors occur when a program does not conform to the grammar of a programming language, and the compiler cannot compile the source file

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

* Sample Program #1 is an example of “Syntax Error” because it had grammar error on line 7 and 17. Line 7 and 17 did not conform to the grammar of a programming language, and the complier was not able to compile the source file
* They had error like not closing the brackets and to not put apostrophes.

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

import turtle

myPen = turtle.Turtle()

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

def drawCircle(rgb) :

**#** myPen.down(

myPen.down()- **Error= bad input (basically needed a closing bracket after 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])

drawCircle(‘circleColours’[circleNumber]) **– Error= circleColours was not defined (basically needed apostrophes ‘’)**

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.

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

* Sample Program #2 is an example of “Run-Time Error” because the error in this sample occurs when the program is running thus, making the program an example of “Run- time Error”.

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

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

* Syntax error is an error that stops the program from running whereas Runtime error is an error where the program starts running but gets errors midway.

**Level 3: Logic Errors**

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

* A logic error (or logical error) is a mistake in a program's source code that results in incorrect or unexpected behavior
* Logic errors occur when the program runs without crashing, but produces an incorrect result. The error is caused by a mistake in the program's logic. You won't get an error message, because no syntax or runtime error has occurred.

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

* Sample Program #3 is an example of a “Logic Error” because the programs run smoothly and does not give an error message. But, instead of printing three circles it prints two and instead of printing colored circles it prints black circles.

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
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 therefore, can’t get the program to run whereas, Logic errors occur when a program does not do what the programmer expects it to do but runs smoothly.
* Syntax error usually gives you an error message whereas Logic error does not.

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

* Run time error is an error that occurs while the program is running and therefore stops the program mid-way and does not finish its job whereas Logic error is an error where the program does not give you any error throughout the program and runs smoothly but fails to give you the end results you wanted .

**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)

CODE:

import turtle

myPen = turtle.Turtle()

myPen.forward(40

#myPen.forward(40)- Needed a closing bracket(syntax error)

myPen.left(90)  
myPen.forward(80)

myPen.right(90)

myPen.forward(80)

myPen.left(90)

myPen.forward(40)

myPen.left(90)

myPen.forwar(80)

#myPen.forward(80) – forward was spelled incorrectly and needed a d and therefore the program stopped midway(runtime-error)

myPen.right(90)

myPen.forward(80)

myPen.left(90)

myPen.forward(40)

myPen.left(90)

myPen.forward(70)

#myPen.forward(80)- the perimeter of the line was a wrong amount and therefore the results were not what I wanted but did not get any error message (logic error)

myPen.right(90)

myPen.forward(80)

myPen.left(90)

myPen.forward(40)

myPen.left(90)

myPen.forward(80)

myPen.right(90)

myPen.forward(80)

**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])