

Unit Plan: Introduction of Python

Topic of the Lesson: Iteration

Grade and Content: 10th, 11th, and 12th grade, CSP

Learning Objectives

1. Students will be able to use “while” & “for” to program in Python.
2. Students will be able to include iteration code segment to solve certain problem.

NYS standards:

9-12.CT.1: Create a simple digital model that makes predictions of outcomes.

9-12.CT.4, 9-12.CT.5, 9-12.CT.6, 9-12.CT.7, 9-12.CT.8, 9-12.CT.9, 9-12.CT.10
Algorithms and Programming

Content-specific vocabulary: iteration

Materials/Resources:

<https://books.trinket.io/pfe/05-iterations.html>

Assessments:

Journal/ programming log

Warm-up/ Mini lesson:

(#1) Students will read the aim and teacher will review the idea of iteration.
Question: Give the names of the iteration block in Snap!.

Turn and Talk (#2): What is the error and how can we fix it?

(#3- 5) Live-coding:

```
n = 5
while n > 0:
    print(n)
    n = n - 1
print('Blastoff!')
```

Focus Questions:

1. Trace the program by using a diagram and predict the output with your partner.

2. Which block do we use in Snap! that is similar to “while” in Python?

(#6 – 7) Teacher will demonstrate “Infinite loops” and how to use “break”.

Activity / Sequence of Lesson:

1. (#8 – 10)

```
friends = ['Joseph', 'Glenn', 'Sally']
for friend in friends:
    print('Happy New Year:', friend)
print('Done!')
```

Focus Questions:

1. Predict the output for this code segment with your partners. Then run the code to confirm.

2. Which block do we use in Snap! that is similar to “for” in Python?

(#11) Teacher will ask students to read the slides about the general instructions how to use “while” and “for” in Python.

(#12-13) Coding Practice: create a program that will count the total number of items in a number list [3, 41, 12, 9, 74, 15] (with scaffolding of pseudocode)

(#14-15) Coding Practice: create a program that will count the sum of the numbers in the list [3, 41, 12, 9, 74, 15] (without pseudocode)

(#16) Students will explain what the program does. (Maximum loop to find the largest number of a list)

(#17) Students will create a minimum loop

(#18-19) Challenge Task:

Write a program which repeatedly reads numbers until the user enters "done". Once "done" is entered, print out the total, count, and average of the numbers. If the user enters anything other than a number, detect their mistake using `try` and `except` and print an error message and skip to the next number.

Code sample:

```
total=0
count=0
a=0
line = ' '
while True:
    line=input('Enter a number: ')
    try:
        line=int(line)
        total = int(total + line)
        count= count+1
        a= total / count
    except:
        if line=='done':
            print (total,count, round(a,2))
```

```
break  
else:  
    print('invalid input')  
    continue
```

Summary / Next Steps / Exit Slip:

Group presentation for the programs. Feedback and comment from the class and the teacher.