1 Physics 29, F23

2 Discussion Session 2, Solutions

3 Exercise 1

[1]: # First way, "brute force", but easy (easier) to understand
inString = "hello world"
outString = ""
# Number of characters, i.e., "length" of string
n = len(inString)
# Loop over string backwards
for i in range(n-1,-1,-1):
    thisCharacter = inString[i]
    outString = outString + thisCharacter
print(outString)
dlrow olleh

[2]: # Second way, very elegant but hard to read
# Stolen from https://www.w3schools.com/python/python_howto_reverse_string.asp
inString = "hello world"
print(inString[::-1])
dlrow olleh

4 Exercise 2

[3]: # First, a brute force method
inString = "hello world"

# put the words into a list
listOfWords = inString.split(" ")

# Number of words, i.e., "length" of the list
n = len(listOfWords)

# Loop over list backwards
outString = ""
for i in range(n-1,-1,-1):
    thisWord = listOfWords[i]
    outString = outString + thisWord
    # Unless we are at the end, should add a blank space
    if i > 0:
        outString = outString + " ">
print(outString)

world hello

[4]: # Now a more elegant (?) method
inString = "hello world"

# put the words into a list
listOfWords = inString.split(" ")

# reverse the list
listOfWords.reverse()

# "join" the list into a string with " " as a separator
print(' '.join(listOfWords))

world hello

5 Exercise 3

5.0.1 see for example https://www.w3schools.com/python/gloss_python_escape_characters.asp

[5]: # new line
print("hello \n world")
print("-------------")

# double slash
print("hello \ world")
print("-------------")

# quote
print("hello \" world")
print("-------------")
print('\n etc. etc.')
hello
world
-------------
hello \ world
-------------
hello " world
-------------

etc. etc.

6  Exercise 4

[6]: # Using a for loop
n = 100
my_sum = 0
for i in range(n+1):
    my_sum = my_sum + i
print(my_sum)
5050

[7]: # Alternative syntax...still with for loop
# For some reason I dont like it and find it confusing
# But most people prefer it to the previous one
n = 100
my_sum = 0
for i in range(n+1):
    my_sum += i
print(my_sum)
5050

[8]: # Turns out that you can actually sum over a list
# Use "range" to make a list
my_list = list(range(0,101))
my_sum = sum(my_list)
print(my_sum)
5050

[9]: # Another way of doing it.
# Using numpy arrays
# (not covered in class yet, a preview)
import numpy as np
arr = np.linspace(1,100,100)  # This is an array of floats now
print("The sum of arr is ",arr.sum())
new_arr = np.linspace(1, 100, 100, dtype='int')  # These are now integers
print("The sum of new_arr is ", new_arr.sum())

The sum of arr is 5050.0
The sum of new_arr is 5050