**CIS-298 Intro to Python**

**With Professor Robert Mann**

**HW #2**

**Student: Demetrius Johnson**

**19 January 2023**

**Due: 24 January 2023 at 4pm**

Submit your code in a report: question number, code copy/pasted, snippet of output.

You may answer all questions in one program but only submit the code for that question, separately from any other code.

\*\*\*\*\*\* Lists has 3 questions, tuples has 3 questions, dictionary has 8 questions. \*\*\*\*\*\*

# Question 1 - Lists

#\* Lists []

#Create an empty list and print the list

my\_list = []

print(my\_list)

#Create a singleton list and print the list

my\_list\_1 = [1]

#Create a list of 5 items of mixed types and print the list

my\_list\_5 = [5, -5.5, 'item', 'c', 0]

print(my\_list\_5)

#Print the 3rd item in the list

print(my\_list\_5[2])

#Print the item at index -3

print(my\_list\_5[-3])

#Change the 3rd item in the list to “bye” and print the whole list

my\_list\_5[2] = "bye"

print(my\_list\_5)

#Change the -4th item in the list to ‘hello’ and print the whole list

my\_list\_5[-4] = "hello"

print(my\_list\_5)

#Print the length of the list

print(len(my\_list\_5))

#Find the min and max of the list

#print(min(my\_list\_5)) --> error, requires list to have elements all of the same type

#print(max(my\_list\_5)) --> error, requires list to have elements all of the same type

#Delete the -5th item in the list

my\_list\_5.pop(-5)

#Add list [‘heaven’, -986] to the beginning of your list

my\_list\_5 = ['heaven', -986] + my\_list\_5

print(my\_list\_5)

#Append list [‘abc’,1,”ABC” ] to the end of your list

my\_list\_5.append(['abc', 1, 'ABC'])

print(my\_list\_5)

#Add ‘hello’ to the end of your list. What happened? 🡪 added ‘hello’ to end of list as a new element

my\_list\_5.append('hello')

print(my\_list\_5)

#Append “world” to the end of your list. What happened? 🡪 added ‘world’ to end of list as a new element

my\_list\_5.append("world")

print(my\_list\_5)

#Print your list, perform pop() on your list, and print the list again

print(my\_list\_5)

my\_list\_5.pop()

print(my\_list\_5)

#Perform pop(4) on your list and print the list

my\_list\_5.pop(4)

print(my\_list\_5)

#Perform pop(-2) on your list and print the list

my\_list\_5.pop(-2)

print(my\_list\_5)

#Print the length of your list as a float

print(float(len(my\_list\_5)))

#Print the type and ord of your list

print(type(my\_list\_5))

#print(ord(my\_list\_5)) error--> ord() function requires a single character

## Screenshot Output

A picture containing text

Description automatically generated

# Question 2 – Tuples

#\* Tuples ()

print("TUPLES")

#Create an empty tuple and print the tuple

empty\_tuple = ()

print(empty\_tuple)

#Create a singleton tuple and print the tuple

singleton\_tuple = (1)

print(singleton\_tuple)

#Create a tuple of 5 items of mixed types and print the tuple

my\_tuple\_5 = (5, -5.5, 'item', 'c', 0)

print(my\_tuple\_5)

#Print the 3rd item in the tuple

print(my\_tuple\_5[2])

#Print the item at index -3

print(my\_tuple\_5[-3])

#Change the 3rd item in the tuple to “bye” and print the whole tuple

# my\_tuple\_5[2]= 'bye' #ERROR: NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

print(my\_tuple\_5)

#Change the -4th item in the tuple to ‘hello’ and print the whole tuple

#my\_tuple\_5[2]= 'hello' #ERROR: NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

print(my\_tuple\_5)

#Print the length of the tuple

print(len(my\_tuple\_5))

#Find the min and max of the tuple

#print(min(tmy\_tuple\_5)) ERROR: MIN AND MAX REQUIRES ALL ELEMENTS OF THE SAME TYPE

#print(max(tmy\_tuple\_5)) ERROR: MIN AND MAX REQUIRES ALL ELEMENTS OF THE SAME TYPE

#Delete the -5th item in the tuple

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Add tuple [‘heaven’, -986] to the beginning of your tuple

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Append tuple [‘abc’,1,”ABC” ] to the end of your tuple

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Add ‘hello’ to the end of your tuple. What happened?

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Append “world” to the end of your tuple. What happened?

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Print your tuple, perform pop() on your tuple, and print the tuple again

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Perform pop(4) on your tuple and print the tuple

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Perform pop(-2) on your tuple and print the tuple

#NO FUNCTION TO REMOVE OR ADD ELEMENTS --> TUPLES ARE IMMUTABLE

#Print the length of your tuple as a float

print(float(len(my\_tuple\_5)))

#Print the type and ord of your tuple

print(type(my\_tuple\_5))

#print(ord(my\_tuple\_5)) error--> ord() function requires a single character

## Screenshot Output

Calendar

Description automatically generated with low confidence

# Question 3 – Dictionaries

#\* Dict { }

#Build a dictionary with 6 college & mascot associations, using all three methods

# A = { key : value, key : value }

Dictionary1 = {"UnivMI":"Wolverines",

"MiState":"Spartans",

"CentralMI":"Chippewas",

"WesterMI":"Broncos",

"NorthernMI":"WildCats",

"EasternMI":"Eagles"

}

# A = dict( [(key, value), (key,value)] )

Dictionary2 = dict([("UnivMI","Wolverines"),

("MiState","Spartans"),

("CentralMI","Chippewas"),

("WesterMI","Broncos"),

("NorthernMI","WildCats"),

("EasternMI","Eagles")]

)

# A=dict( key=value, key=value )

Dictionary3 = dict(UnivMI = "Wolverines",

MiState = "Spartans",

CentralMI = "Chippewas",

WesterMI = "Broncos",

NorthernMI = "WildCats",

EasternMI = "Eagles"

)

print(Dictionary1, Dictionary2, Dictionary3)

#Access the dictionary using a key that doesn’t exist

#print(Dictionary1["FloridaState"]) -->exception thrown --> key not in dictionary

#Add a new value to the dictionary and print the dictionary

Dictionary1["FloridaState"] = "Seminoles"

print(Dictionary1)

#Change the value of an existing entry and print the dictionary

Dictionary1["FloridaState"] = "Alligators"

print(Dictionary1)

#Del an entry from the dictionary

del Dictionary1["FloridaState"]

print(Dictionary1)

#Check to see if a value is ‘in’ the dictionary and also ‘not in’ the dictionary

print ('Wolverines' in Dictionary1.values())

print ('Wolverines' not in Dictionary1.values())

#Get the length of the dictionary

print(len(Dictionary1))

#Print a list of sorted keys to the dictionary

print(sorted(Dictionary1))

## Screenshot Output

Text

Description automatically generated