

Assignment 6

June 7, 2023

1 Assignment 6

1.0.1 Sue Susman, MEd, N

1.0.2 Date

1.1 Question 1

- 1) import the random library.
- 2) Use `random.seed(10)` to initialize a pseudorandom number generator.
- 3) Create a list of 50 random integers from 0 to 15. Call this list `int_list`.
- 4) Print the 10th and 30th elements of the list.

You will need to use list comprehension to do this. The syntax for list comprehension is: `[<expression> for <item> in <iterable>]`. For this question your expression will be a randint generator from the random library and your iterable will be `range()`. Research the documentation on how to use both functions.

```
[1]: # your code here
import random

random.seed(10)
int_list = [random.randint(0, 15) for _ in range(50)]

print("10th element:", int_list[9])
print("30th element:", int_list[29])
```

```
10th element: 1
30th element: 7
```

2 Question 2

- 1) import the string library.
- 2) Create the string `az_upper` using `string.ascii_uppercase`. This is a single string of upper-case letters

- 3) Create a list of each individual letter from the string. To do this you will need to iterate over the string and append each letter to the an empty list. Call this list `az_list`.
- 4) Print the list.

You will need to use a for-loop for this. The syntax for this for-loop should be:

```
for i in string>:    <list operation>
```

```
[2]: # your code here
import string

az_upper = string.ascii_uppercase
az_list = []

for letter in az_upper:
    az_list.append(letter)

print(az_list)
```

```
['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P',
'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']
```

2.1 Question 3

- 1) Create a set from 1 to 5. Call this `set_1`.
- 2) Create a set from `int_list`. Call this `set_2`.
- 3) Create a set by finding the `symmetric_difference()` of `set_1` and `set_2`. Call this `set_3`.
- 4) What is the length of all three sets?

```
[3]: # your code here
set_1 = set(range(1, 6))
set_2 = set(int_list)
set_3 = set_1.symmetric_difference(set_2)

length_set_1 = len(set_1)
length_set_2 = len(set_2)
length_set_3 = len(set_3)

print("Length of set_1:", length_set_1)
print("Length of set_2:", length_set_2)
print("Length of set_3:", length_set_3)
```

```
Length of set_1: 5
Length of set_2: 15
Length of set_3: 12
```

2.2 Question 4

- 1) Import default dict and set the default value to 'Not Present'. Call this dict_1.
- 2) Add int_list, set_2, and set_3 to dict_1 using the object names as the key names.
- 3) Create a new dictionary, dict_2, using curly bracket notation with set_1 and az_list as the keys and values.
- 4) Invoke the default value of dict_1 by trying to access the key az_list. Create a new set named set_4 from the value of dict_1['az_list']. What is the length of the difference between dict_2['az_list'] and set_4?
- 5) Update dict_2 with dict_1. Print the value of the key az_list from dict_2. What happened?

```
[6]: # your code here
from collections import defaultdict

dict_1 = defaultdict(lambda: 'Not Present')
dict_1['int_list'] = int_list
dict_1['set_2'] = set_2
dict_1['set_3'] = set_3

dict_2 = {'set_1': set_1, 'az_list': az_list}

set_4 = set(dict_1['az_list'])
difference_length = len(set(dict_2['az_list']).difference(set_4))

dict_2.update(dict_1)
print("Value of 'az_list' in dict_2:", dict_2['az_list'])
```

Value of 'az_list' in dict_2: Not Present