FEU Institute of Technology

MACHINE PROBLEM

3

Type of Collection

<Justine Guillermo>
<W22>
<February 5, 2020>

A. Given the output below, write a swift syntax that will generate each output:

1. How many elements are there in set A and B

2. How many elements are there in B that is not part of A and C

3. [h, i, j, k]

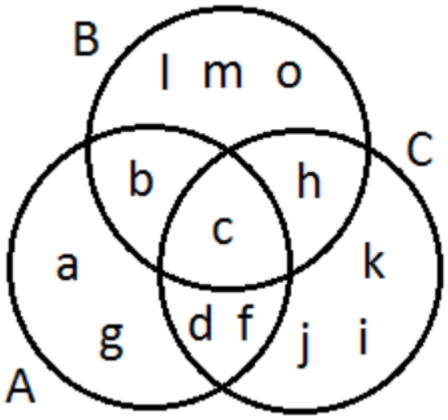
4. [c, d, f]

5. [b, c, h]

6. [d, f]

7. [c]

8. [l, m, o]



B. Given a collection of items with their price. Write a program that will simulate the output

Apple: 15, Orange:16, Mango:25, Banana:7, Strawberry: 35, Ponkan:6

Total Number of Items: 6

Items: Apple, Orange, Mango, Banana, Strawberry, Ponkan

Sorted Items: Apple, Banana, Mango, Orange, Ponkan, Strawberry

Searched Item: Apple

Item: Apple with a price of 15

Items Selection:

Banana(3) 21

Mango(4) 100

Strawberry(1) 35

Total Price: 156 (8 items)

Source Code A:

```
 #A
setA = ['a', 'b', 'c', 'd', 'f', 'g']
```

```
setB = ['b', 'c', 'h', 'l', 'm', 'o']
setC = ['c', 'd', 'f', 'j', 'h', 'i', 'k']

andb = setA + setB
print('How many elements are there in set A and B: ', len(set(andb)))

andc = setA + setC
if setB not in andc:
    val = (list(set(setB).difference(set(andc))))
print('How many elements are there in B that is not part of A and C: ', len(val))

if setC not in andb:
    val2 = (list(set(setC).difference(set(andb))))

num3 = list(set(setB).intersection(setC)) + list(set(val2))
num3.remove('c')

num4 = list(set(setA).intersection(setB, setC)) + \
    list(set(setA).intersection(setC))

num5 = list(set(setA).intersection(setB, setC)) + \
    list(set(setA).intersection(setB)) + list(set(setB).intersection(setC))

num6 = list(set(setA).intersection(setC))
num6.remove('c')

print(sorted(list(set(num3))))
print(sorted(list(set(num4))))
print(sorted(list(set(num5))))
print(sorted(list(set(num6))))
print(list(set(setA).intersection(setB, setC)))
print(sorted(list(val)))
```

Output A:

```
How many elements are there in set A and B:  10
How many elements are there in B that is not part of A and C:  3
['h', 'i', 'j', 'k']
['c', 'd', 'f']
['b', 'c', 'h']
['d', 'f']
['c']
['l', 'm', 'o']

Process finished with exit code 0
```

Source Code B:

```
#B
thisisdict = {"Apple": 15, "Orange": 16,  "Mango": 25,  "Banana": 7,  "Strawberry":
```

```
35, "Ponkan": 6}
print(thisisdict)
print("Total Number of Items: ", len(thisisdict))
print("Items: ", end='')
for x in thisisdict:
    print(x ,end=' ')
print("\nSorted Items: ", end='')
for x in sorted(thisisdict):
    print(x ,end=' ')
print()
search = 'Apple'
print(f'Searched Item: {search}')
if search in thisisdict:
    print('Items: ',search,' with a price of ',thisisdict.get(search))
print('Items Selection: ')
item = 'Banana'
quan = 3
print(item,f'({quan})', quan*thisisdict.get(item))
```

Output B:

```
{'Apple': 15, 'Orange': 16, 'Mango': 25, 'Banana': 7, 'Strawberry': 35, 'Ponkan': 6}
Total Number of Items:  6
Items: Apple Orange Mango Banana Strawberry Ponkan
Sorted Items: Apple Banana Mango Orange Ponkan Strawberry
Searched Item: Apple
Items:  Apple  with a price of  15
Items Selection:
Banana (3) 21

Process finished with exit code 0
```