S02_T01_Estructura de dades

January 25, 2022

1 IT Academy - Data Science Itinerary

1.1 S02-T01: Data structure

1 :

Create a list that groups the months of the year into quarters (Q1: January, February and March, Q2: April, May, June …), that is, a list with 4 lists inside.

[1]:
```
months = [['January', 'February', 'March'],
            ['April', 'May', 'June'],
            ['July', 'August', 'September'],
          ['October', 'November', 'December']]
```

2:

Create a code that allows you to acces: + 2.1. the second month of the first quarter. + 2.2 the months of the first quarter. + 2.3 september and october.

- 2.1. the second month of the first quarter.

[2]:
```
m2_q1 = months[0][1]

print(m2_q1)
```

```
February
```

- 2.2 the months of the first quarter.

[3]:
```
q1_all = months[0]

print(q1_all)
```

```
['January', 'February', 'March']
```

[4]:
```
for month in q1_all:
    print(month)
```

```
January
February
March
```

- 2.3 september and october.

[5]:
```
a = "September"
a_month_index = []
for quarter in months:
    if a in quarter:
        i = months.index(quarter)
    for month in quarter:
        if a in month:
            j = quarter.index(month)
            a_month_index.append(quarter.index(month))

print(i,j) # i,j is used to find the index of values in a list "months"
print(months[i][j])
```

```
2 2
September
```

[6]:
```
a = "October"

for quarter in months:
    if a in quarter:
        i = months.index(quarter)
    for month in quarter:
        if a in month:
            j = quarter.index(month)
            a_month_index.append(quarter.index(month))
print(i,j)
print(months[i][j])
```

```
3 0
October
```

3:

Create a list of disordered numbers and answer the following questions:

- 3.1 How many numbers are there?
- 3.2 How many times does the number 3 appear
- 3.3 How often do the numbers 3 and 4 appear?
- 3.4 What is the largest number?
- 3.5 What are the 3 smallest numbers?
- 3.6 What is the range of this list?

- 3.1

We can use the sample() method available in random module to directly generate a list of random numbers

[7]:
```
import random

randomlist = random.sample(range(1, 100), 30)
print(randomlist)
```

```
len(randomlist)   # returns the number of items of an object.
```

```
[60, 63, 51, 49, 76, 88, 13, 68, 83, 54, 66, 27, 53, 50, 98, 75, 56, 97, 30, 35,
69, 95, 19, 62, 7, 79, 84, 10, 71, 43]
```

[7]:
```
30
```

- 3.2 We can use the count() method to returns the number of elements with the specified value.

[8]:
```
print(randomlist.count(3))
```

```
0
```

- 3.4 With the max() function we can find the largest item in an list

[9]:
```
print(max(randomlist))
```

```
98
```

- 3.5 Using sorted() function the list in ascending order and print the 3 smallest element in the list

[10]:
```
print(sorted(randomlist)[0:3])
```

```
[7, 10, 13]
```

- 3.6 the range difference between the smallest and highest numbers in a list or set.

[11]:
```
range =(max(randomlist)-min(randomlist))
print(range)
```

```
91
```

4 :

Create a dictionary as follows:

compra = {“Pomes”: {“Qty”: 5, “€”: 0.42}, “Peres”: {“Qty”: 3, “€”: 0.66}}

and answer the questions:

- 4.1 Add some more fruit
- 4.2 How much did the pears cost in total?
- 4.3 How many fruits did we buy in total?
- 4.4 What is the most expensive fruit?

- 4.1:

[12]:
```
compra = {"Pomes": {"Qty": 5, "€": 0.42}, "Peres": {"Qty": 3, "€": 0.66}}
print(compra)
```

```
{'Pomes': {'Qty': 5, '€': 0.42}, 'Peres': {'Qty': 3, '€': 0.66}}
```

[13]:
```
compra["Bananas"]= {"Qty": 3, "€": 0.43}
compra["Uvas"]= {"Qty": 2, "€": 0.55}
compra["Melocoton"]= {"Qty": 5, "€": 0.52}
```

[14]:
```
print(compra)
```

```
{'Pomes': {'Qty': 5, '€': 0.42}, 'Peres': {'Qty': 3, '€': 0.66}, 'Bananas':
{'Qty': 3, '€': 0.43}, 'Uvas': {'Qty': 2, '€': 0.55}, 'Melocoton': {'Qty': 5,
'€': 0.52}}
```

- 4.2

[15]:
```
cost = compra["Peres"]["Qty"]*compra["Peres"]["€"]
print(cost)
```

```
1.98
```

- 4.3

[16]:
```
total = 0

for k, v in compra.items():
     total += v["Qty"]
print(total)
```

```
18
```

- 4.4

[17]:
```
fruit_price = []

for k, v in compra.items():
    fruit_price.append(v["€"])

print(max(fruit_price)) #the most expensive price
```

```
0.66
```

[18]:
```
for k, v in compra.items():
    if v["€"] == max(fruit_price):
        print(k) #the key of the most expensive price
```

```
Peres
```

[ ]:
```

```