Python - Data Analysis Essentials

Main Exercises, Day 1

Giuseppe Accaputo

g@accaputo.ch

Universität Zürich[UZH]

IT Training and Continuing Education

Exercises on Data Structures

Warm-Up: Getting to Know Lists

- Define a function that takes a list as an argument
- The function first checks the length of the list: The list should contain at least 3 elements
- If the number of element is correct, the function replaces the first and third element with the value `0`
- The function will print each element on a new line

- Example run:

CODE

```
list = [1,2,3,4]
change_list(list)
```

INTERPRETER

OUTPUT

```
0
2
0
4
```

Warm-Up: Getting to Know Dictionaries

- We want to create a menu list
- Define a dictionary that uses the meal as the key (string) and the price as the corresponding value (float)
- Insert some meals into the dictionary
- Now define a function `print_menu` that will take this dictionary as its argument and prints the menu list (including the prices)

- Example run:

CODE

```
menu = {
    …
}

print_menu(menu)
```

INTERPRETER

OUTPUT

```
Our menu:
 * Burger, 10.5 CHF
 * Fries, 4.0 CHF
 * Chicken Nuggets, 8.25 CHF
```

Count the Occurences of a Single Character in a Word

- Write a function `count_occur(char, word)` which returns the number of times the single character `char` occurs within the string `word`
- *Hint*: We need a counter to keep track of the number of times we already have encountered the character `char` in `word`

- Function calls and expected return values:

Function call	Return value
`count_occur("a", "Halleluja")`	`2`
`count_occur("e", "Mount Everest")`	`2`
`count_occur("k", "Cathedral")`	`0`

Remove the Borders

– Write a function `middle` that takes a list as its argument and returns a new list, with the first and last element removed

– Function calls and expected return values:

Function call	Return value
`middle([1,2,3,4])`	`[2,3]`
`middle([1,3,2])`	`[3]`
`middle([3,2,5,20,5,100])`	`[2,5,20,5]`

Computing the Average Out of Multiple Numbers

- Write a function `average` which takes a list of numbers as its argument and returns the average value out of all numbers in the list
- *Note*: The list can contain an arbitrary number of elements

- Function calls and expected return values:

Function call	Return value
`durchschnitt([1,2,3,4])`	`2.5`
`durchschnitt([4,18,30,-20])`	`8.0`
`durchschnitt([3,3,3,3])`	`3.0`

Largest and Smallest Number in a List

- Write a function `min_max` which takes a list of numbers as its argument and returns a tuple consisting of exactly two elements: the first element is the smallest element in the list and the second element is the biggest element in the list
  - *Hint*: `min(list)` returns the smallest element in `list`, whereas `max(list)` will return the biggest element in list

- Function calls and expected return values:

Function call	Return value
`min_max([102,-2,30,400])`	`(-2,400)`
`min_max([-123,430,5000,-300])`	`(-300,5000)`

A List of Lists

– The following list of lists is given:

CODE

```
employees = [["Marco", "Sales", 1428], ["Javier", "Customer Care", 859],
["Giuseppe", "Engineering", 891]]
```

– Write a program that based on the given list of lists outputs the following on the screen:

OUTPUT

```
* Marco works in Sales (Employee-Nr.: 1428)
* Javier works in Customer Care (Employee-Nr.: 859)
* Giuseppe works in Engineering (Employee-Nr.: 891)
```

Export the Employee List to CSV

- Write a program that exports the following list of lists containing all employees to a proper CSV file:

```
employees = [["Marco", "Sales", 1428], ["Javier", "Customer Care", 859],
["Giuseppe", "Engineering", 891]]
```

- Once the CSV file has been written by your program, try to import it to Excel and see if it converted everything successfully
  - The resulting CSV file should consist of 3 rows, with each row having 3 cells

Stock Management

– A company just sent you the CSV file `current_stock.csv` containing information about the current stock. The data is ordered as follows: Product name, value, current stock, date of the last sale

`current_stock.csv`

```
TV,1000,145,4/5/2018
Computer,2000,19,10/10/2018
iPad,400,200,9/12/2018
iPhone,800,8,11/11/2018
```

– Your task now is to provide the company with a program that warns the company if an item is low on stock, namely when the stock value is below `20` for a given item

OUTPUT

```
Warnings:
 * Computer is low on stock: only 19 items left
 * iPhone is low on stock: only 8 items left
```