Assignment #2: Simple data

Master in Informatics and Computing Engineering
Programming Fundamentals
Instance: 2018/2019

0. Introduction

***Goals**: to write programs using variables, operations using simple data, input and output.*

***Pre-requirements (prior knowledge)**: Basic data and operations as in Lecture #2*

***Rules**: You may work with colleagues, however, each student must write and submit in Moodle his or her this assignment separately. Be sure to indicate with whom you have worked. We may run tools to detect plagiarism (e.g.duplicate code submitted).*

***Deadline**: 8:00 Monday of the week after (08/10/2018)*

***Collaborators**:*

list here their codes (2018nnnn1, 2018nnnn2, …)

1. Variables and print

Take the sentence: "*The quick brown fox jumps over the lazy dog*".

Use Spyder3 to create a file called **`fox.py`** in your "Working space" (see Assignment #1).

Store each word of the sentence in a separate variable and then print the sentence using the variables you created.

Then, run the program and copy the result followed by your program (**`fox.py`**) here:

```
# copy of the result


# copy of the program
```

2. Input and string operators

Use Spyder3 to create a file called **`html.py`** and write a program that asks a user to input a tag **`h`** and a string **`text`** and prints an HTML valid element of the form *<tag>text</tag>*.

For example, for a user input tag *h1* and a text "*I'm an HTML text*", the output must be "*<h1>I'm an HTML text</h1>*"

Then, run the program for a user input tag *b* and *html's* as text, and copy the result followed by your program (the content of the file **`html.py`**) here:

```
# copy of the result

# copy of the program
```

3. Input and cast

Use Spyder3 to create a file called **`cast.py`** and write a program that asks a user to input a number **`n`**, calculates the expression **`n + nn + nnn`** and prints its value.

For example, for a user input 5, the output must be 5 + 55 + 555 = 615.

Then, run the program for a user input **9**, and copy the result followed by your program (the content of the file **`cast.py`**) here:

```
# copy of the result

# copy of the program
```

4. Operations and operands

The formula for computing the final amount if one is earning compound interest is given on Wikipedia as:

$$A = P\left(1 + \frac{r}{n}\right)^{nt}$$

Where:

- P = principal amount (the amount that the interest is provided on)
- r = the interest rate
- n = the frequency that the interest is paid out (per year)
- t = the number of years that the interest is calculated for

Use Spyder3 to create a file called **`interests.py`** and write a program that replaces these letters with something a bit more human-readable, and calculate the final amount (A) at the end of the second year, for some varying amounts of money (P) at realistic interest rates (r):

1. P = 1000, n = 2, and r = 1%
2. P = 650, n = 1 and r = -0.05%
3. Values of P, n and r introduced by the user

Then, run the program, and copy the result followed by your program (the content of the file **`interests.py`**) here:

```
# copy of the result


# copy of the program
```

5. Remainder

Use Spyder3 to create a file called **`alarm_clock.py`** and write a program that determines what is the actual time (using **`now`** of class **`datetime`** from module **`datetime`**) and prints it in the form *hh:mm* (hours and minutes).

Given that an alarm is set up for *8 hours and 30 minutes* later, the program prints the time on display at the time of the alarm, in the same format.

Then, run the program, and copy the result followed by your program (the content of the file **`alarm_clock.py`**) here:

```
# copy of the result
Now: 22:45
Alarm: 07:15

# copy of the program
```

The end.

FPRO, 2018/19