Exercise 2

due 09.11.2017

This exercise covers Chapters 0-2 of the tutorial. Please submit your solution until 09.11., 23:59, via e-mail to `programming-l1-ws1718@ims.uni-stuttgart.de` as a **plain text** and/or Python file (which should end on `.txt` or `.py`). Please also submit in groups of **at least 3 students**, and clearly indicate the **names and immatriculation** numbers of all involved students. Submissions that do not fulfill these requirements are not accepted.

Questions

1. The following code represents an attempt of collecting information about someone. All are supposed to be variable assignments.

```
first name = "Nils"
street == "Pfaffenwaldring"
house = 5b
zip 70569
city = Stuttgart"
yearOfBirth=1980
```

- How many errors are there? What are they?
- Correct all errors you found
- Assuming your corrections, please assign an additional variable `ads`. This variable controls whether the poor person will receive advertisement, it should be of a boolean value. A person gets advertisement if they are between 25 and 65 (excluding) of age and live in zip codes below 10000 or above 60000 (including). We assume that there is a variable `currentYear`, set to 2017.

2. In this exercise, you will extend the calculator you have written in the last exercise in the following ways (you may do all at once or step by step).

- In addition to being asked for two numbers, the user is now also asked for a math operation they want to do (either `+` or `*`). The program should then either add or multiply the numbers.

- The program also accepts the names of the math operations (`add` or `addition` and `multi` or `multiplication`).
- Instead of exiting the program after calculation, the program asks the user whether they want to continue. If the user enters some form of confirmation, the program starts over. If not, it exits.
- If the result of the calculation is between 5 (excluding) and 10 (including), the program should print the number as a word (e.g., `six` instead of `6`).

The listing below contains an example run

```
Enter number 1:
5
Enter number 2:
6
Enter operation:
*
Result: 30
Do you want to continue (Y/N):
Y
Enter number 1:
4
Enter number 2:
3
Enter operation:
add
Result: seven
Do you want to continue (Y/N):
N
```

3. You have been asked to write code to be used in an ATM. The code you write is responsible for deciding how many of which kind of banknotes are given out. The following banknotes exist (the same list currently in use by the Euro): 5, 10, 20, 50, 100, 200, 500. The program you write asks the user for a number and prints the number of bills of each kind that are returned. If it is not possible to give out bills amounting to the given value, an error message is printed. The program immediately asks for the next value. Banknotes should be as high as possible.

   The following shows a possible run

```
Enter amount you want to withdraw:
15
5: 1
10: 1
Enter amount you want to withdraw:
55
```

```
5: 1
50: 1
Enter amount you want to withdraw:
45
5: 1
20: 2
```