Homework 2:
Object-oriented Programming

Benjamin Roth, Marina Sedinkina
Symbolische Programmiersprache

Due: Thursday November 1, 2018, 12:00 (noon)

In this exercise you will:

- Practice creating simple classes and objects with Python.
- As always, you need to submit your solution using the folder we provided in your GitLab account.

Exercise 1: Bank account class [5 points]

1. Using the slides & the script, put together a file containing the complete Account class. Each method must have a documentation string at the beginning which describes what the method is doing.
2. Create a main application where you create a number of accounts. Play around with depositing / withdrawing money. Change the account holder of an account using a setter method.
3. Change the withdraw function such that the minimum balance allowed is -1000.
4. Write a function `apply_interest(self)` which applies an interest rate of 1.5% to the current balance and call it on your objects.
5. Draw a UML diagram representing your Account class. **Submit the UML diagram as a file called `oop_1_bank.pdf` in the same directory as the python file `oop_1_bank.py`** Hint: There is an editor called 'dia' which makes it easy to create UML diagrams. It is available for Linux, MacOS and Windows (`http://dia-installer.de/download/index.html`).

Exercise 2: Employee class [4 points]

1. Write the complete code for the Employee class (including constructor, `__str__`,...)
2. Create a few employee objects and show how you can manipulate them using the methods.
3. Draw a UML class diagram for your Employee class. **Submit the UML diagram as a file called `oop_2_employee.pdf` in the same directory as the python file `oop_1_employee.py`**.