Assignment 4

Grading

Mandatory requirements (all assignments):

1. OOP paradigms
2. Classes of maximum 200 lines (except the UI classes)
3. Methods of maximum 30 lines
4. Java naming conventions https://google.github.io/styleguide/javaguide.html

Minimal requirements for accepting the assignment (5 points):

- Implement the class diagram from the homework specification. Choose wisely the appropriate data structures for saving the Persons and the Accounts. Differentiate between the Map and the Set data structures.
- Graphical interface:
  - Window for Person operations: add new Person, edit Person, delete Person, view all Persons in a table (JTable)
  - Window for Account operations: add new Account, edit Account, delete Account, view all Accounts in a table (JTable)
- Differentiate between the saving and the spending accounts. The saving account allows a single large sum deposit and withdrawal and computes an interest during the deposit period. The spending account allows several deposits and withdrawals, but does not compute any interest.
- Add/Withdraw money from the accounts
- Documentation

Requirements for maximum grade (5 points):

Requirement	Grading
Use JTables for display. Implement click listeners on the JTable.	1 point
Design a test driver for the system (JUnit)	1 point
Design by contract: preconditions and postconditions in the BankProc interface. Implement them in the Bank class using the assert instruction. Define the invariant for the class Bank.	1 point
Design Pattern Observer for notifying the account holder when one of its accounts is modified	1 point
Save the information from the Bank class in a file using serialization. Load the information when the application starts.	1 point

Submission

- Create a repository on *bitbucket* with the name: *PT2017_Group_Name_Assigment_4*
- Push the source code and the documentation (push the code not an archive with the code)
- Share the repository with the user *utcn_dsrl*

Tutorials

- Java Serialization
  - http://www.tutorialspoint.com/java/java_serialization.htm
- Java HashMap
  - http://javarevisited.blogspot.ro/2011/02/how-hashmap-works-in-java.html
- Java assert
  - http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html
  - http://javarevisited.blogspot.ro/2012/01/what-is-assertion-in-java-java.html
  - http://stackoverflow.com/questions/11415160/how-to-enable-the-java-keyword-assert-in-eclipse-program-wise