Contents

[Banking System 2](#_Toc117782768)

[Overview 2](#_Toc117782769)

[Banking accounts’ types 2](#_Toc117782770)

[First: basic BankAccount 2](#_Toc117782771)

[Second: SavingsBankAccount 2](#_Toc117782772)

[Client class 2](#_Toc117782773)

[The main class (BankingApplication) 2](#_Toc117782774)

[Main class options 3](#_Toc117782775)

[Create a new account 3](#_Toc117782776)

[List clients and accounts 3](#_Toc117782777)

[Withdraw money 3](#_Toc117782778)

[Deposit money 3](#_Toc117782779)

# Banking System

## Overview

* Allow to create back account for a specific client
* User account can
  + Display his account details
  + Withdraw money
  + Deposit money
* There are two types of this banking accounts
* There is a Client class.
* The main class that runs the application

## Banking accounts’ types

### First: basic BankAccount

* Holds the following data
  + Account ID
  + Balance
* The following method apply to this class
  + Two different constructors:
    - The first sets the balance to a given value
    - The second has no argument sets the balance to 0
  + Setters and getters to access private data
  + Withdraw function
    - Withdraws an amount of money from the account if the balance is sufficient.
  + Deposit
    - It deposits an amount of money in the account.

### Second: SavingsBankAccount

* It extends the basic BankAccount and may have some extra data fields and operations.
* This account requires the user to keep a minimum amount of money in the account, which is called the minimum balance, as long as the account is open.
* It also requires him to make deposits that are not less than 100 a time
* So this class has additional data fields
  + minimumBalance: This minimum balance takes a default value of 1000 L.E.
  + Constructor: sets the value of the initial balance and the minimum balance
    - Initial balance should be >= min balance.
  + Setters and getters to access the private data
  + Withdraw function: It overrides the method withdraw to allow withdrawing money but not below the minimum balance.
  + Deposit: It deposits an amount of money in the account but only if the amount is >= 100 LE.

## Client class

It holds:

* The basic information of a client like his name, address and phone number.
* A pointer to his bank account.
* An account also points to its owner.

## The main class (BankingApplication)

* This class displays the main menu and accepts the user's choice.
* It maintains a list of accounts and clients.
* It allows the user to perform operations on a bank account.

## Main class options

### Create a new account

Clients must enter the following information:

* His name
* His address
* His phone number
* Type of bank account he would like
* The starting balance in his bank account

At the end we should show him:

* His ID
* His current balance

### List clients and accounts

We should print all the clients and their bank account information

* His name
* His address
* His phone number
* His ID
* The type of his back account
* The current balance in his bank account

### Withdraw money

* User should first enter his account ID
* We should show him his account information
  + His ID
  + His name
  + The type of his bank account
  + The current balance in his bank account
* User should enter the amount to withdraw

At the end we should show him:

* Print: Thank you.
* His account ID
* The new balance in his bank account

### Deposit money

* User should first enter his account ID
* We should show him his account information
  + His ID
  + His name
  + The type of his bank account
  + The current balance in his bank account
* User should enter the amount to deposit

At the end we should show him:

* Print: Thank you.
* His account ID
* The new balance in his bank account