

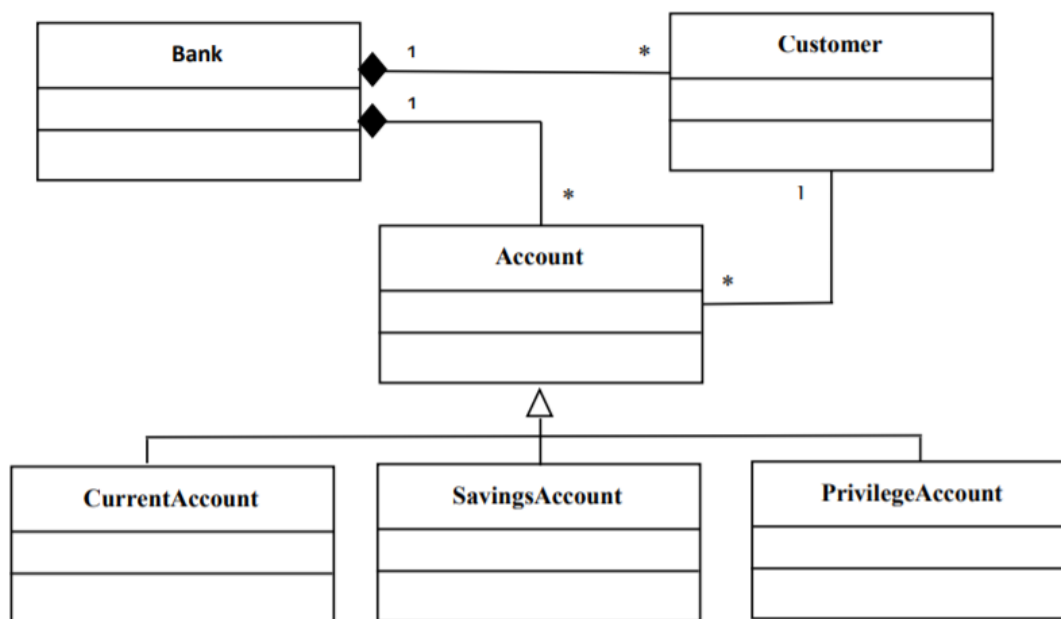


BANK SYSTEM

ДОКУМЕНТАЦИЯ ЗА ПРОЕКТ BANK SYSTEM,
ИЗГОТВЕН ЗА КУРСА КАЧЕСТВЕН ПРОГРАМЕН КОД,
II СЕМЕСТЪР, 2018/2019 АКАДЕМИЧНА ГОДИНА

Радост Дрънгова, ФН: 62209,
спец. Софтуерно инженерство,
2-ри курс

Проекта представлява проста банкова система, реализираща следната йерархия:



В изградената система е предвидено да има единичен екземпляр на класа Bank, в който се съдържат много екземпляри на класовете Account и Customer. В останалата част от този документ ще получите подробна информация за всички класове и методите, които те използват:

1. Customer (потребител)

Член-данни:

- Unsigned long long id – идентификатор на потребителя
- string name – име на потребителя
- string address – адрес на потребителя

Методи:

- Customer конструктор с параметри и без параметри
- селектори: getId, getName, getAddress
- void display – извежда информация за потребителя

2. Account (банкова сметка)

Член-данни:

- string iban – номер на банкова сметка (IBAN)
- unsigned long long ownerId – идентификатор на потребителя, собственик на банковата сметка
- double amount – налична сума в сметката

Методи:

- Account конструктор с параметри;
- Account конструктор без параметри;
- void чисто виртуален метод deposit – за добавяне на сума към банковата сметка
- bool чисто виртуален метод withdraw – за изтегляне на сума от банковата сметка, ако това е възможно
- void чисто виртуален метод display – за извеждане на информация за сметката
- double getBalance – връща наличната сума в сметката
- string getIBAN – връща IBAN на сметката

- unsigned long long getOwnerId – връща ID на собственика на сметката

2.1 CurrentAccount (текуща сметка)

Класът CurrentAccount наследява Account и реализира методите:

- void deposit – за добавяне на сума към банковата сметка
- bool withdraw – за изтегляне на сума от банковата сметка. Ако в сметката има по-малко пари от исканата сума, връща false, иначе – намалява сумата на сметката с исканата сума и връща true
- void display – за извеждане на информация за сметката – вид на сметка, IBAN, номер на потребител, баланс

2.2 SavingsAccount (спестовна сметка)

Класът SavingsAccount наследява Account като го допълва с

- double interestRate – годишен лихвен процент

и реализира методите:

- double getInterestRate – връща годишния лихвен процент
- void deposit – за добавяне на сума към банковата сметка
- bool withdraw – за изтегляне на сума от банковата сметка. Ако в сметката има по-малко пари от исканата сума, връща false, иначе – намалява сумата на сметката с исканата сума и връща true
- void display – за извеждане на информация за сметката – вид на сметка, IBAN, номер на потребител, годишния лихвен процент, баланс

2.3 PrivilegeAccount (привилегирована сметка)

Класът PrivilegeAccount наследява Account като го допълва с

- double overdraft – позволен овърдрафт (сума превишаваща кредита)

и реализира методите:

- double getOverdraft – връща позволения овърдрафт
- void deposit – за добавяне на сума към банковата сметка
- bool withdraw – за изтегляне на сума от банковата сметка. Ако наличната сума в сметката + позволения овърдрафт е по-малко от исканата сума, връща false, иначе – намалява сумата на сметката с исканата сума и връща true (в този случай може да се получи отрицателно число за баланса)
- void display – за извеждане на информация за сметката – вид на сметка, IBAN, номер на потребител, позволен овърдрафт, баланс

3. Bank (банка)

Член-данни:

- String name – име на банката
- string address – адрес на банката
- vector <Customer> customers – списък от потребители (изберете подходящ начин за представяне)
- vector <Account*> accounts – списък от банкови сметки (изберете подходящ начин за представяне)

Методи:

- Bank конструктор с параметри и без параметри
- селектори: getName, getAddress

- void addCustomer с параметри customerId, name и address– ако не съществува потребител с идентификатор customerId, създава нов потребител и го добавя към списъка с потребителите
- void listCustomers – извежда списък с потребителите
- void deleteCustomer с параметър customerId – изтрива потребител с идентификатор customerId от списъка с потребители, както и всички негови сметки
- void addAccount с параметри accountType, iban, ownerId, amount – създава нова сметка от съответния тип и я добавя към списъка с банкови сметки. Ако не съществува потребител с идентификатор ownerId или съществува банкова сметка със същия IBAN – извежда съобщение за грешка.
- void deleteAccount с параметър iban – изтрива сметка с съответен IBAN
- void listAccounts – извежда информация за всички сметки в банката
- void listCustomerAccount с параметър customerId – извежда информация за банковите сметки на потребител с идентификатор customerId
- void transfer с параметри fromIBAN, toIBAN и amount – извършва банков превод на сума amount от банкова сметка с идентификатор fromIBAN към банкова сметка с идентификатор toIBAN, ако е възможно
- void display – извежда информация за банката и колко потребителя и сметки има

Всички методи са реализирани и без параметри за да могат да премат такива от стандартния вход.

4. Main

В главната програма се създава банка и да се реализира следното меню:

- 1 List customers
- 2 Add new customer
- 3 Delete customer
- 4 List all accounts
- 5 List customer accounts
- 6 Add new account
- 7 Delete account
- 8 Withdraw from account
- 9 Deposit to account
- 10 Transfer
- 11 Display info for the bank
- 12 Quit