class Customer:

The class represents a customer.

property:
private String name;
private String password;
private String email;
private String address;

private CheckingAccount checkingAccount;
private SavingAccount savingAccount;

private boolean accountOpenFee;
private int currentMonth;
private int freeTransferChance;
private String operatingDate;
private ArrayList<Loan>loans;

---

class Banker:

The class represents a banker;

Property:
private String name;
private String password;

---

class Account:

The class represents a generic class for a checking account and a saving account.

property:
protected Map<String, String> account;

function:
public Account()
public Account(Map<String, String> account)
public void setBalanceByKey(String key, String value)
public String getBalanceByKey(String key)
public void addBalanceByKey(String key, String increment)
public void deductBalanceByKey(String key, String decrement)

---

class CheckingAccount:

The class extends the Account class and represents a checking account of a customer.

---

class SavingAccount:

The class extends the Account class and represents a saving account of a customer.

---

class Loan:

The class represents a loan.

property:
private int index;
private String applicationTime;
private String collateral;
private String loanAmount;
private String debt;

function:
public Loan(int index, String applicationTime, String collateral, String loanAmount)
public Loan(int index, String applicationTime, String collateral, String loanAmount, String debt)
public void setApplicationTime(String applicationTime)
public void setCollateral(String collateral)
public void setLoanAmount(String loanAmount)
public void setDebt(String debt)
public int getIndex()
public String getApplicationTime()
public String getCollateral()
public String getLoanAmount()
public String getDebt()
public void repay(String decrement)
public void addInterest(String increment)

class Transaction:

The class represents any transaction in the system.

property:
private String date;
private String name;
private String type;
private String detailedContent;
private String fee;

function:
public Transaction(String date, String name, String type, String detailedContent, String fee)
public String getDate()
public String getName()

public String getType()
public String getDetailedContent()
public String getFee()

class jsonRW:

The class is designed to read and write json files which store data of costumers and the banker.

property:

private String customersJsonFilePath;
private String bankerJsonFilePath;
private String transactionsJsonFilePath;
private File customersJsonFile;
private File bankerJsonFile;
private File transactionsJsonFile;

function:

public JsonRW()
public String readJsonFile(String filePath)
public void writeJsonFile(String filePath, String jsonString)
public ArrayList<String> getCustomersNames()
public Customer getCustomerByName(String name)
public Banker getBanker()
public ArrayList<Transaction> getAllTransactions()
public void updateCustomers(Customer customer)
public void updateTransactions(Transaction transaction)

class NumberValidation:

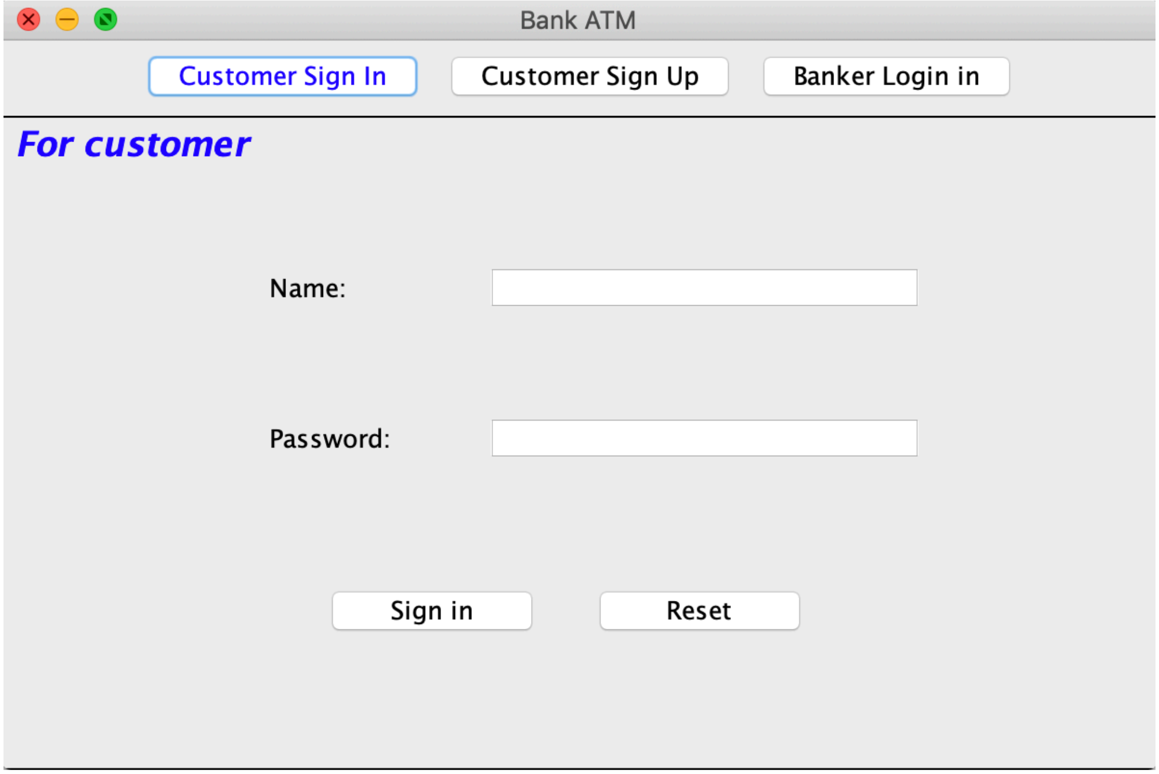
The class is designed to check the validity of the input number.

function:

public boolean isDepositNumberValid(String number)
public boolean isWithdrawalNumberValid(String number)
public boolean isTransferNumberValid(String number)
public boolean isLoanNumberValid(String number)

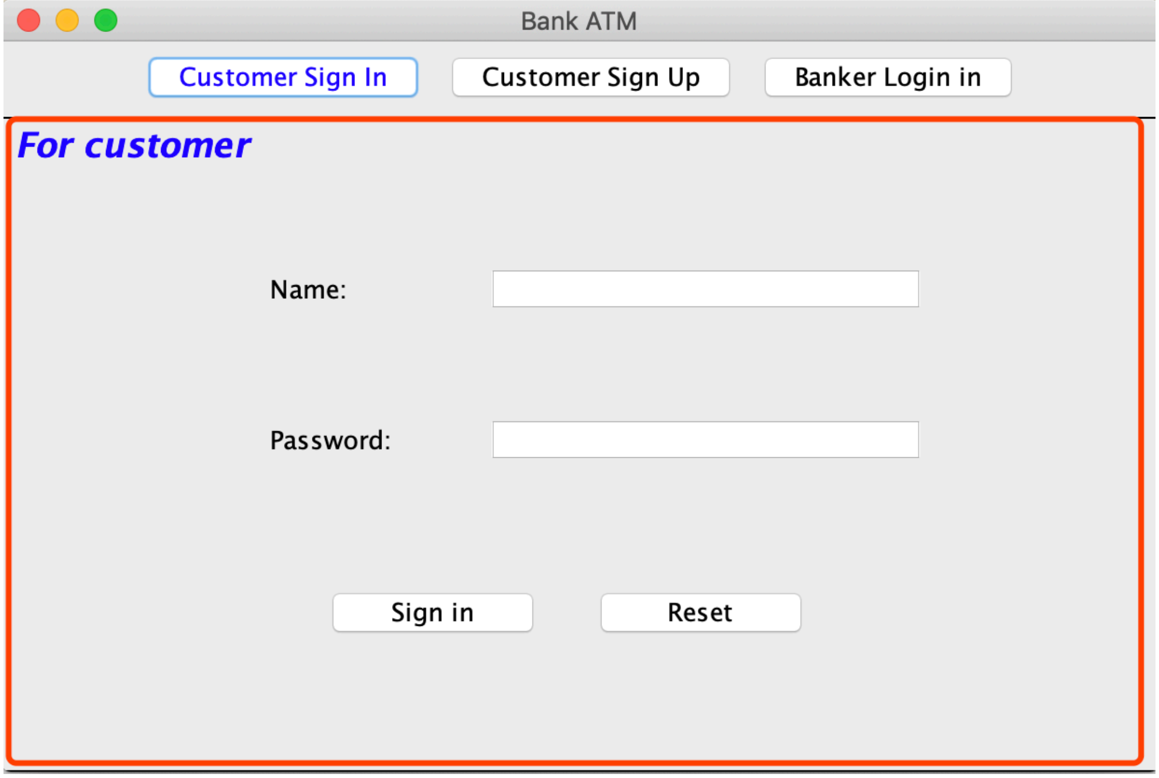
class SignIn:

The class is used to control the interface of signing up and signing in. It uses a cardLayout.



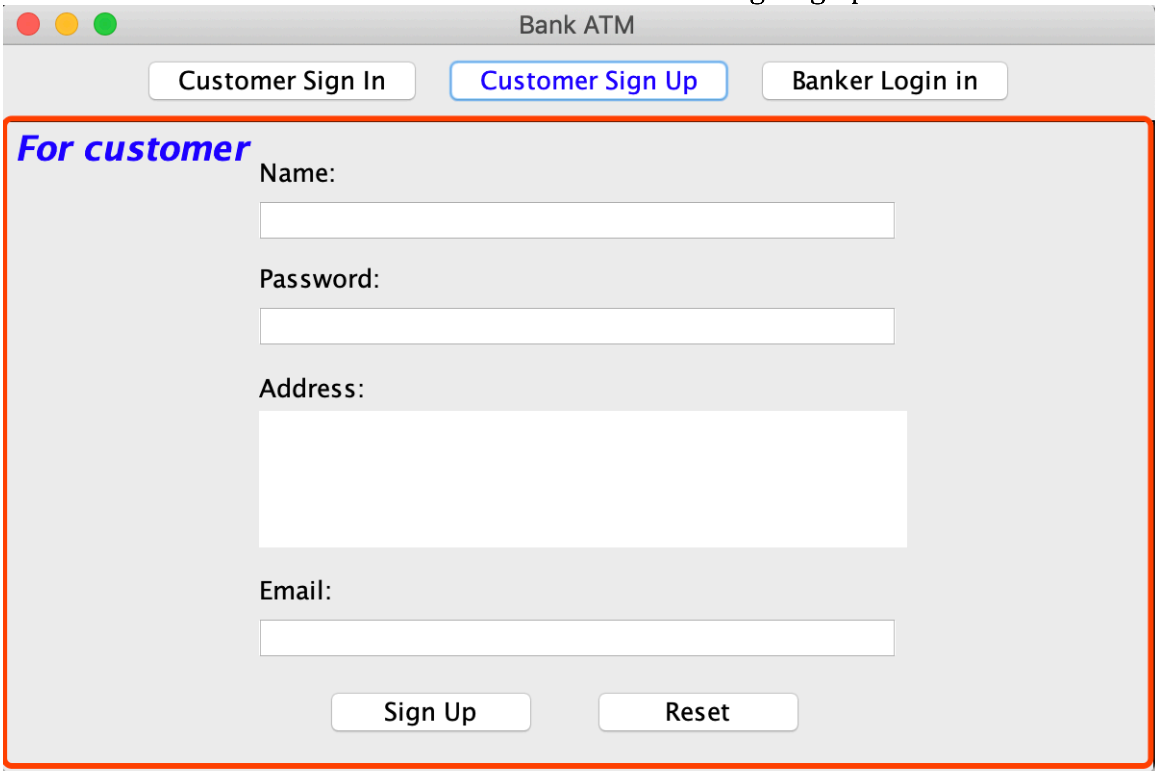
class CustomerSignIn:

The class is used to control the interface of customer signing in.



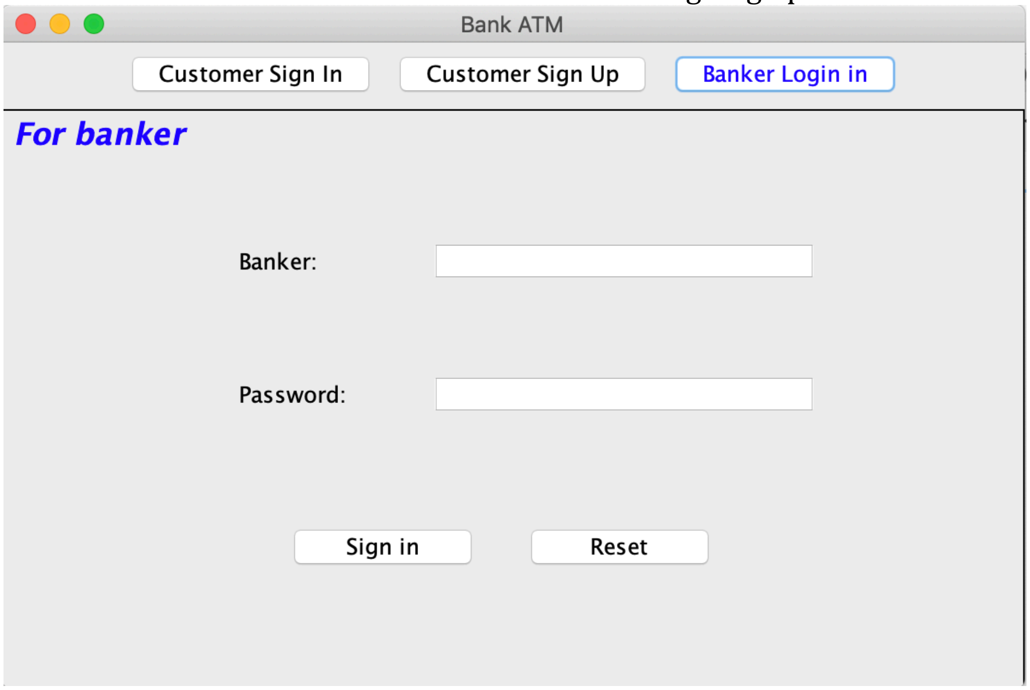
class CustomerSignUp:

The class is used to control the interface of customer signing up.



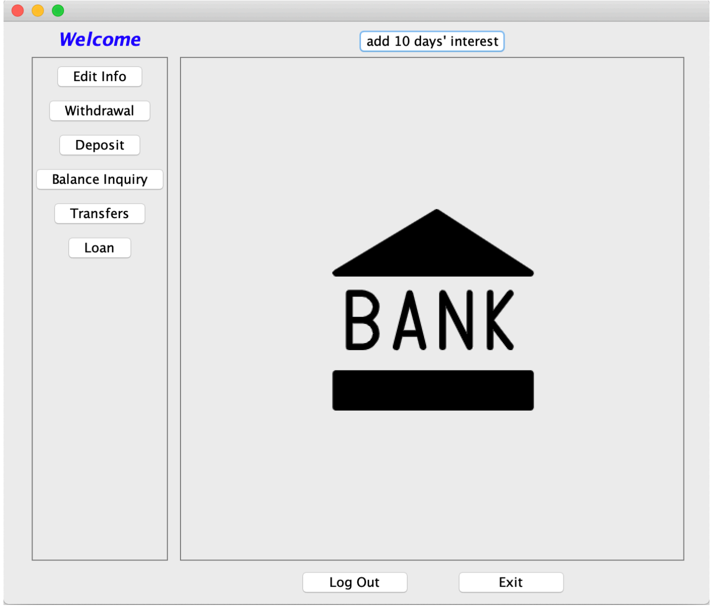
class BankerSignUp:

The class is used to control the interface of customer signing up.



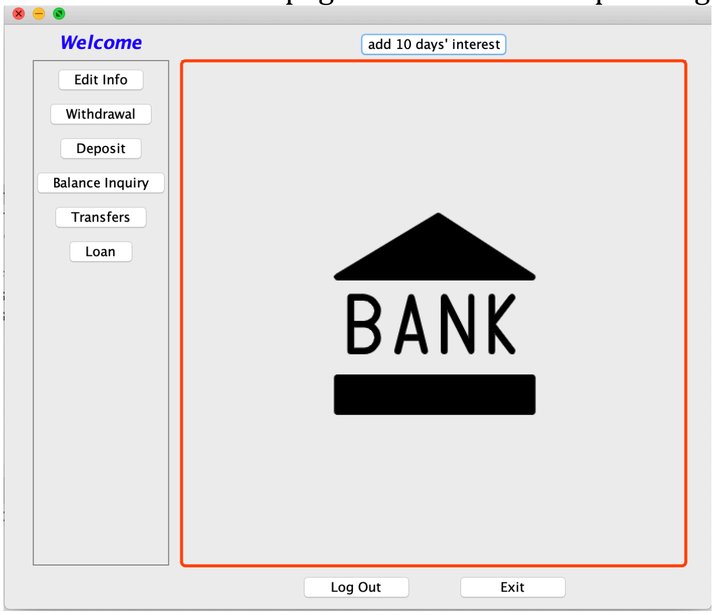
class CustomerFrame:

The class is used to control the overall arrangement of the customer operating interface. It uses a cardLayout.



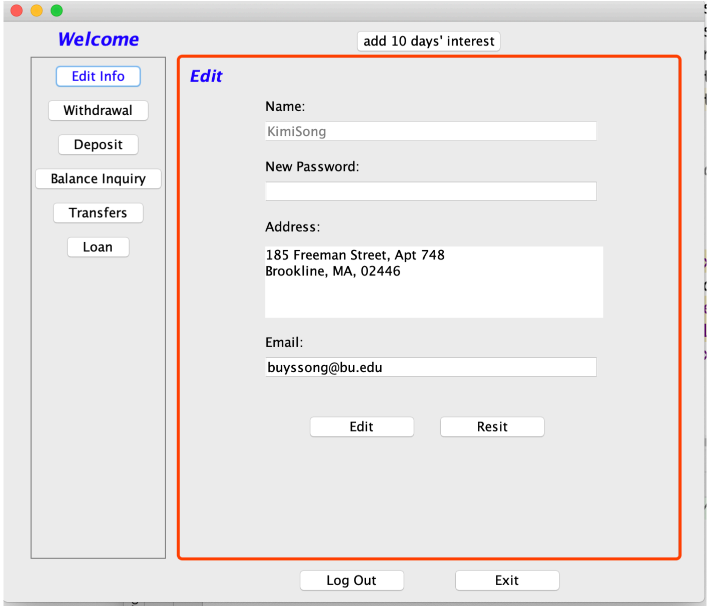
class HomePage:

The class is used to control the homepage of the customer operating interface.



class Edit:

The class is used to control the panel of edit.



Edit

Name:
KimiSong

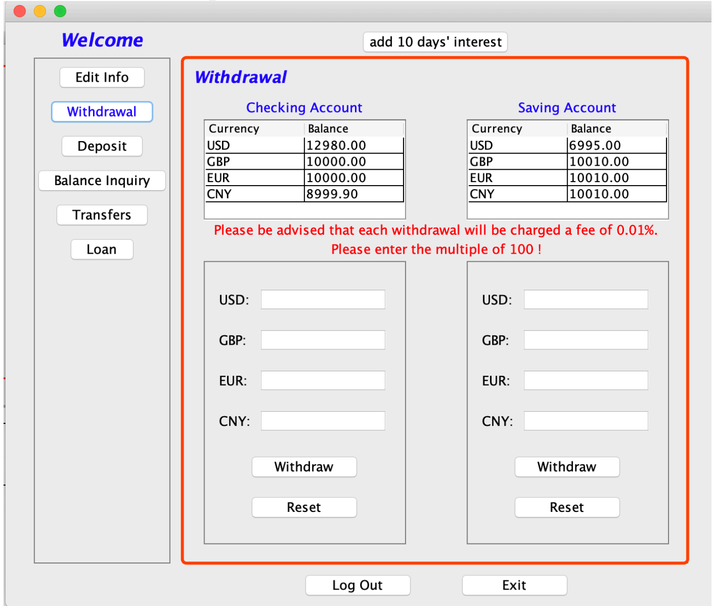
New Password:

Address:
185 Freeman Street, Apt 748
Brookline, MA, 02446

Email:
buyssong@bu.edu

class Withdrawal:

The class is used to control the panel of withdrawal.



Withdrawal

Checking Account

Currency	Balance
USD	12980.00
GBP	10000.00
EUR	10000.00
CNY	8999.90

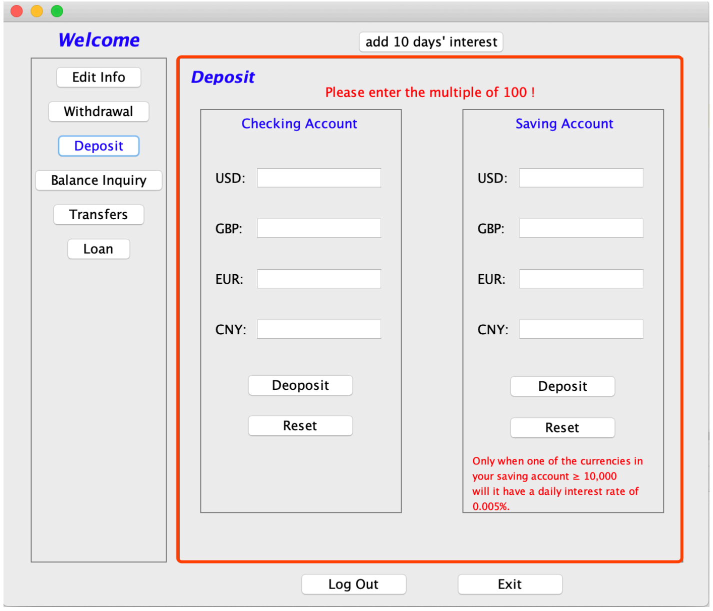
Saving Account

Currency	Balance
USD	6995.00
GBP	10010.00
EUR	10010.00
CNY	10010.00

Please be advised that each withdrawal will be charged a fee of 0.01%.
Please enter the multiple of 100 !

class Deposit:

The class is used to control the panel of deposit.

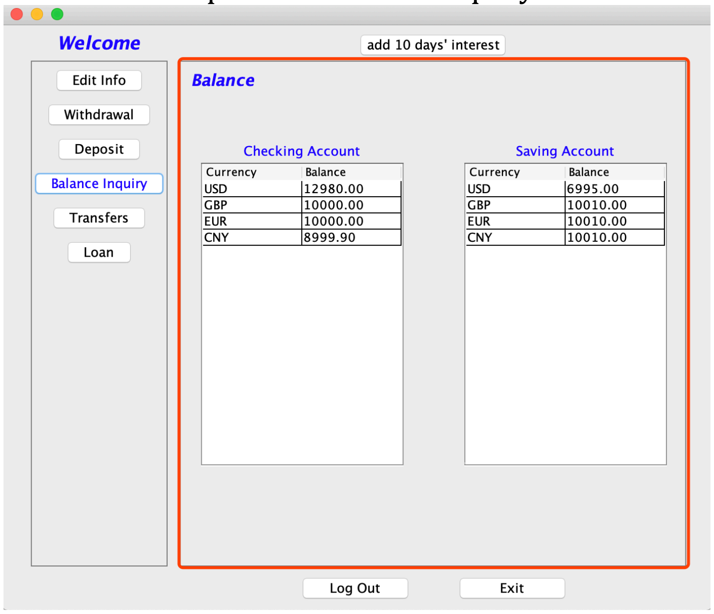


Please enter the multiple of 100 !

Only when one of the currencies in
your saving account ≥ 10,000
will it have a daily interest rate of
0.005%.

class Balance:

The class is used to control the panel of balance inquiry.



Checking Account

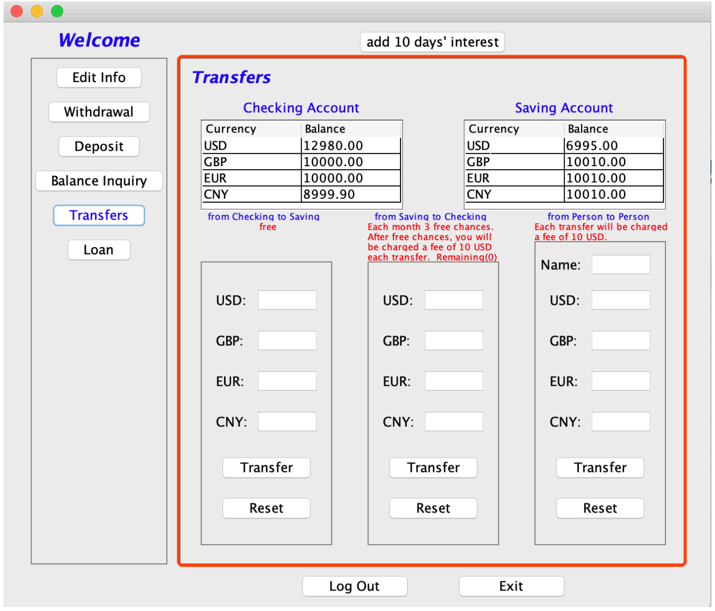
Currency	Balance
USD	12980.00
GBP	10000.00
EUR	10000.00
CNY	8999.90

Saving Account

Currency	Balance
USD	6995.00
GBP	10010.00
EUR	10010.00
CNY	10010.00

class Transfers:

The class is used to control the panel of transfers.



Checking Account

Currency	Balance
USD	12980.00
GBP	10000.00
EUR	10000.00
CNY	8999.90

Saving Account

Currency	Balance
USD	6995.00
GBP	10010.00
EUR	10010.00
CNY	10010.00

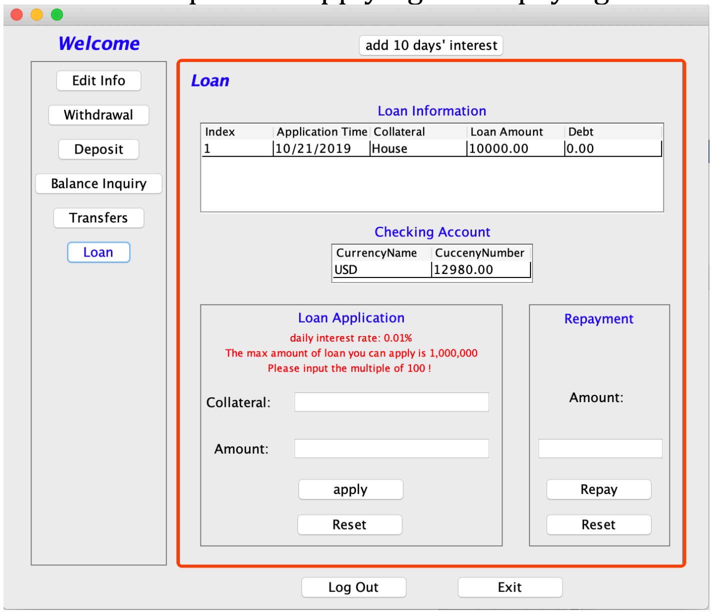
from Checking to Saving
free

from Saving to Checking
Each month 3 free chances.
After free chances, you will
be charged a fee of 10 USD
each transfer. Remaining(0)

from Person to Person
Each transfer will be charged
a fee of 10 USD.

class LoanPanel:

The class is use to control the panel of applying and repaying a loan.



Loan Information

Index	Application Time	Collateral	Loan Amount	Debt
1	10/21/2019	House	10000.00	0.00

Checking Account

CurrencyName	CucchenyNumber
USD	12980.00

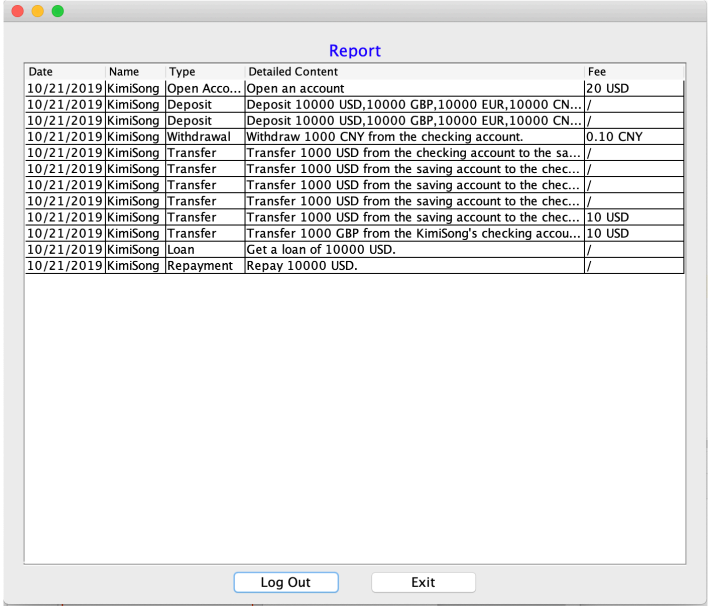
Loan Application

daily interest rate: 0.01%
The max amount of loan you can apply is 1,000,000
Please input the multiple of 100 !

Repayment

class BankerFrame:

The class is used to control the banker opearing interface.



Report

Date	Name	Type	Detailed Content	Fee
10/21/2019	KimiSong	Open Acco...	Open an account	20 USD
10/21/2019	KimiSong	Deposit	Deposit 10000 USD,10000 GBP,10000 EUR,10000 CN...	/
10/21/2019	KimiSong	Deposit	Deposit 10000 USD,10000 GBP,10000 EUR,10000 CN...	/
10/21/2019	KimiSong	Withdrawal	Withdraw 1000 CNY from the checking account.	0.10 CNY
10/21/2019	KimiSong	Transfer	Transfer 1000 USD from the checking account to the sa...	/
10/21/2019	KimiSong	Transfer	Transfer 1000 USD from the saving account to the chec...	/
10/21/2019	KimiSong	Transfer	Transfer 1000 USD from the saving account to the chec...	/
10/21/2019	KimiSong	Transfer	Transfer 1000 USD from the saving account to the chec...	/
10/21/2019	KimiSong	Transfer	Transfer 1000 USD from the saving account to the chec...	10 USD
10/21/2019	KimiSong	Transfer	Transfer 1000 GBP from the KimiSong's checking accou...	10 USD
10/21/2019	KimiSong	Loan	Get a loan of 10000 USD.	/
10/21/2019	KimiSong	Repayment	Repay 10000 USD.	/