Account
overdraftLimit double
creationDate Date
users List<String>
transactions List<Transaction>
accountID String
balance double
Account(String, Date, double)
getID() String
increaseBalance(double) void
decreaseBalance(double) void
getBalance() double
setBalance(double) void
getCreationDate() Date
addUser(String) void
ownedBy(String) boolean
addTransaction(Transaction) void
canWithdraw(double) boolean
canDeposit(double) boolean
canTransferOut(double) boolean
withdraw(double) boolean
deposit(double) boolean
isPrimary() boolean
recordBalance() void
getBalanceString() String
update() void
getTransactionsList() List<Transaction>
getCurrencySymbol() char
Transaction
primary Account
amount double
date Date
revertible boolean
executable boolean
reverted boolean
alreadyRequested boolean
Transaction(Account, double, Date)
isAlreadyRequested() boolean
setAlreadyRequested(boolean) void
isExecutable() boolean
setExecutable(boolean) void
isRevertible() boolean
setRevertible(boolean) void
isReverted() boolean
setReverted(boolean) void
getDate() Date
canExecute() boolean
canRevert() boolean
performExecution() void
performReversion() void
recordTransaction() void
execute() boolean
revert() boolean
transactionInfo() String
toString() String
DebtAccount
DebtAccount(String, Date)
getBalanceString() String
AssetAccount
AssetAccount(String, Date, double)
getBalanceString() String
ForeignCurrencyAccount
currencyCode String
currencySymbol Character
currencyPermanentlySet boolean
ForeignCurrencyAccount(String, Date)
setCurrencyCode(String) boolean
getCurrencyCode() String
getCurrencySymbol() char
getBalanceString() String
toString() String
TransferTransaction
secondary Account
currencyExchange CurrencyExchange
TransferTransaction(Account, Account, double, Date, CurrencyExchange)
canExecute() boolean
canRevert() boolean
performExecution() void
performReversion() void
convertedAmountOriginalTypeToForeign() double
convertedAmountSecondaryTypeToForeign() double
recordTransaction() void
transactionInfo() String
BillPaymentTransaction
outgoingFileName String
payee String
BillPaymentTransaction(Account, double, Date, String)
canExecute() boolean
canRevert() boolean
performExecution() void
performReversion() void
updateOutgoing() void
transactionInfo() String
DepositTransaction
DepositTransaction(Account, double, Date)
canExecute() boolean
canRevert() boolean
performExecution() void
performReversion() void
transactionInfo() String
WithdrawalTransaction
atm ATM
WithdrawalTransaction(Account, double, Date, ATM)
canExecute() boolean
canRevert() boolean
performExecution() void
performReversion() void
transactionInfo() String
CreditCardAccount
CreditCardAccount(String, Date)
canTransferOut(double) boolean
toString() String
LineOfCreditAccount
LineOfCreditAccount(String, Date)
toString() String
SavingsAccount
SavingsAccount(String, Date)
compoundInterest() void
update() void
toString() String
ChequingAccount
primary boolean
ChequingAccount(String, Date)
isPrimary() boolean
setPrimary(boolean) void
toString() String
CashableGIC
investment int
interestRate double
CashableGIC(String, Date)
payInterest() void
update() void
toString() String
TransferToUserTransaction
username String
TransferToUserTransaction(Account, Account, double, Date, String, CurrencyExchange)
transactionInfo() String
ATM
billQuantityLimit int
billMinimumLimit int
calculator ATMCalculator
alertFileName String
depositsFileName String
fives int
tens int
twenties int
fifties int
totalDollarAmount int
ATM()
getAlertFileName() String
withdraw(int) boolean
needsRestock() boolean
withdraw(int, int[]) void
withdrawalMessage(int, int, int, int, int) String
add(int, int) void
canWithdraw(int) boolean
getNewTotalDollarAmount(int) int
alertRestock() void
getFives() int
getTens() int
getTwenties() int
getFifties() int
getTotalDollarAmount() int
getRIQuantityLimit() int
stockLevel() String
getDeposits() ArrayList<String[]>
FinanceProduct
associated Account
interestRate double
duration int
username String
initialInfo String
creationDate Date
investment double
PaymentDue double
FinanceProduct(String, int, double, Date)
setAssociated(Account) void
getUsername() String
getAssociated() Account
getInvestment() double
getDuration() int
setInterestRate(double) void
getInitialInfo() String
setInitialInfo(String) void
setInitialInfo() void
getCreationDate() Date
getPaymentDue() double
UpdateDurations() void
UpdateInvestment() void
MakePaymentDue() void
initializable() boolean
StopPaymentDue() void
update() void
AccountManager
accountS HashMap<String, Account>
accountsCurrencyExchange CurrencyExchange
AccountManager()
addAccount(Account) void
getAccountOfID(String) List<String>
getPrimaryAccountID(String) String
associateProductAccount(FinanceProduct) void
payBill(String, double, Date, String) boolean
deposit(String, double, Date) boolean
withdraw(String, int, Date, ATM) boolean
transfer(String, String, double, Date) boolean
transferToUser(String, String, double, Date) boolean
getAccountString(String) String
getAccountSummaryString(String) String
getAccountOwnedSummaryString(String) String
updateAll() void
getTransactionBytes(String, int) Transaction
getRecentAccountTransactions(String, int) List<Transaction>
getTotalBalance(String) double
setCurrencyType(String, String) boolean
getListOfForeignCurrencyAccounts(String) List<ForeignCurrencyAccount>
getAccountsCurrencyExchange() CurrencyExchange
addUser(String, String) void
Request
username String
accountManager AccountManager
productManager ProductManager
Request(String)
getUsername() String
setAccountManager(AccountManager) void
setProductManager(ProductManager) void
execute() boolean
RevertTransactionRequest
transaction Transaction
RevertTransactionRequest(String, Transaction)
isRequested() boolean
setTransactionAlreadyRequested() void
isValid() boolean
execute() boolean
toString() String
AccountRequest
accountType String
accountFactory AccountFactory
AccountRequest(String, String, AccountFactory)
execute() boolean
toString() String
ProductRequest
productType String
productFactory ProductFactory
duration int
investment double
ProductRequest(String, String, int, double, ProductFactory)
execute() boolean
toString() String
ATMCalculator
atm ATM
ATMCalculator(ATM)
canWithdrawBasicUnit() boolean
canWithdrawAdvanced(int) int[]
findWithdrawalSolutions(int, int[], int[], int[], int[]) ArrayList<Integer[]>
computeDenominationCombination(int[], int[]) int
myCopy(int[]) Integer[]
Mortgage
Mortgage(String, int, Date, double, double)
setInitialInfo() void
initializable() boolean
update() void
toString() String
NonRedeemableGIC
NonRedeemableGIC(String, int, Date, double, double)
setInitialInfo() void
initializable() boolean
update() void
toString() String
Role
privilegeNames List<String>
name String
username String
bank Bank
privilegeFactory PrivilegeFactory
privileges List<Privilege>
Role(String, String, Bank, PrivilegeFactory)
getOptionsTree(GraphicalInputReader, MessageDisplay) OptionsTree
setPrivilegeNames() void
generatePrivileges() void
getName() String
RequestManager
accountRequests List<Request>
revertTransactionRequests List<Request>
productRequests List<Request>
accountManager AccountManager
productManager ProductManager
RequestManager(AccountManager, ProductManager)
getAccountRequests() List<Request>
getRevertTransactionRequests() List<Request>
getProductRequests() List<Request>
addRequest(AccountRequest) void
addRequest(ProductRequest) void
addRequest(RevertTransactionRequest) boolean
removeRequest(AccountRequest) void
removeRequest(ProductRequest) void
removeRequest(RevertTransactionRequest) void
removeRequest(int, int) void
getRequest(int, int) Request
executeRequest(int, int) boolean
UserRole
name String
UserRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
EmployeeRole
name String
EmployeeRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
TransactionsRole
name String
TransactionsRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
ATMMaintenanceRole
name String
ATMMaintenanceRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
RequestRole
name String
RequestRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
AccountManagementRole
name String
AccountManagementRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
AdminRole
name String
AdminRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
ViewAccountRole
name String
ViewAccountRole(String, Bank, PrivilegeFactory)
setPrivilegeNames() void
Privilege
name String
username String
atm ATM
accountManager AccountManager
productManager ProductManager
userManager UserManager
requestManager RequestManager
timeManager BankTimeManager
Privilege(String, String, Bank)
performOn(GraphicalInputReader, MessageDisplay) void
getOptionsTree(GraphicalInputReader, MessageDisplay) OptionsTree
selectAccount(GraphicalInputReader, String) String
Package EmployeePrivilege
Package CustomerPrivilege
ChangePasswordPrivilege
ChangePasswordPrivilege(String, Bank)
perform(GraphicalInputReader, MessageDisplay) void
OptionsTree
children List<OptionsTree>
option String
command Command
OptionsTree(String)
OptionsTree()
OptionsTree(String, Command)
executeCommand() void
getOption() String
addOptionsTree() void
addOptionsTree() void
add(String, Command) void
add(int, String, Command) void
hasChildren() boolean
getChildFromUserInput(GraphicalInputReader) OptionsTree
toString() String
Bank
serialVersionUID long
atm ATM
accountManager AccountManager
productManager ProductManager
userManager UserManager
requestManager RequestManager
bankTimeManager BankTimeManager
Bank(ATM, AccountManager, ProductManager, UserManager, RequestManager, BankTimeManager)
getATM() ATM
getUserManager() UserManager
getAccountManager() AccountManager
getProductManager() ProductManager
getRequestManager() RequestManager
getBankTimeManager() BankTimeManager
Session
user User
running boolean
continueProgram boolean
currentOption OptionsTree
optionsTree OptionsTree
inputReader GraphicalInputReader
messageDisplay MessageDisplay
Session(User, GraphicalInputReader, MessageDisplay)
run() boolean
generateOptionsTree() void
exit() void
resetCurrentOption() void
terminateProgram() void
setUser(User) void
CurrencyExchange
myURL String
margin double
supportedCurrencies List<String>
CurrencyExchange()
addSupportedCurrency(String) void
addDefaultAvailableCurrency() void
getSupportedCurrencies() List<String>
getConversionDataObject(String, String) JsonObject
getResultsString(String, String) String
getConversionRateJsonObject() double
getConvertedAmount(double, String, String) double
getAmountAfterMargin(double) double
getActualConvertedAmount(double, String, String) double
Main
frame Frame
inputReader GraphicalInputReader
messageDisplay MessageDisplay
saveFileName String
bank Bank
userFactory UserFactory
main(String[]) void
loadBank() void
initializeFactories() void
initializeDefaultUser() void
saveBank() void
User
username String
password String
roles List<Role>
User(String, String)
addRole(Role) void
getRoles() List<Role>
isValidPassword(String) boolean
setPassword(String) void
getUsername() String
containsRole(String) boolean
BankTimeManager
dayFormat SimpleDateFormat
hourMinuteFormat SimpleDateFormat
customDates SimpleDateFormat
bankDate Date
setDate(Date) void
getDateFormat() SimpleDateFormat
getCurrentTime() Date
updateDate() void
isStartOfMonth() boolean
GraphicalInputReader
frame Frame
GraphicalInputReader(Frame)
getString(String, String) String
getDouble(String, String) double
getInt(String, String) int
getPositiveDouble(String, String) double
getPositiveInt(String, String) int
getSelectionFromOptions(List<?>, String) int
LoginDialog
usernameField JTextField
passwordField JPasswordField
succeeded boolean
LoginDialog(Frame, LoginManager)
getUsername() String
getPassword() String
isSucceeded() boolean
PrivilegeFactory
accountFactory AccountFactory
productFactory ProductFactory
userFactory UserFactory
PrivilegeFactory(AccountFactory, ProductFactory)
setUserFactory(UserFactory) void
getPrivilege(String, String, Bank) Privilege
ProductManager
FinanceProducts List<FinanceProduct>
ProductManager()
addProduct(FinanceProduct) void
updateAll() void
getOwnerProductInfo(String) String
toString() String
ProductFactory
bankTimeManager BankTimeManager
accountManager AccountManager
ProductFactory(BankTimeManager, AccountManager)
getAvailableProductTypes() List<String>
getProduct(String, String, int, double) FinanceProduct
UserFactory
bank Bank
roleFactory RoleFactory
UserFactory(Bank, RoleFactory)
getAvailableUserTypes() List<String>
getUser(String, String, String) User
UserManager
users HashMap<String, User>
UserManager()
addUser(User) void
getUser(String) User
containsUser(String) boolean
AccountFactory
bankTimeManager BankTimeManager
AccountFactory(BankTimeManager)
getAvailableAccountTypes() List<String>
getAccount(String, String) Account
MessageDisplay
frame Frame
MessageDisplay(Frame)
showMessage(String, String) void
showInfoMessage(String, String) void
ButtonList
selection int
ButtonList(Frame, List<?>, String)
getSelection() int
setSelection(int) void
KeyboardInputDialog
inputField JTextField
KeyboardInputDialog(Frame, String, String)
getInput() String
RoleFactory
privilegeFactory PrivilegeFactory
RoleFactory(PrivilegeFactory)
getRole(String, String, Bank) Role
LoginManager
userManager UserManager
LoginManager(UserManager)
login(String, String) boolean
ObjectSerializer
readObject(String) T
writeObject(T, String) void
Command
execute() void