UML DIAGRAMS

group_0308

package phase2.Accounts

AccountManager

- sv: ArrayList<Savings>
- cq: ArrayList<Chequing>
- cc: ArrayList<CreditCard>
- cb: ArrayList<CashBackCard>
- Ic: ArrayList<LineOfCredit>
- owner: User
- getAssetBalance(): doubkle
- getDebtBalance(): double
- getTotalBalance(): double
- + getAccountList(String): ArrayList
- + getTransactions(): ArrayList<Transaction>
- + balanceSummary(): String
- + add(Account): void
- + add(String, String, Calendar): void
- + getAccount(int): Account

CreditCard

accountNum: int

- balance: ForeignCurrency
- owner1: User
- owner2: User
- transactions: ArrayList<Transaction>
- creditLimit: ForeignCurrency
- dateOfCreation: Calendar
- getCreationDate(): String
- + getBalance(): ForeignCurrency
- + setBalance(ForeignCurrency): void
- + getTransactions(): ArrayList<Transaction>
- + helpRead(String): ForeignCurrency
- + helpWrite(ForeignCurrency): void
- + setNewOwner(User): void
- + subtract(Transaction): void
- + subtract(ForeignCurrency): void
- + add(Transaction): void
- + add(String): void
- + getPastTransaction(int): Transaction
- + addTransaction(Transaction): void
- + decreaseCreditLimit(ForeignCurrency): void
- + increaseCreditLimit(ForeignCurrency): void
- + transactionString(): String
- + getAccountNum(): int
- # getCreditLimit(): ForeignCurrency

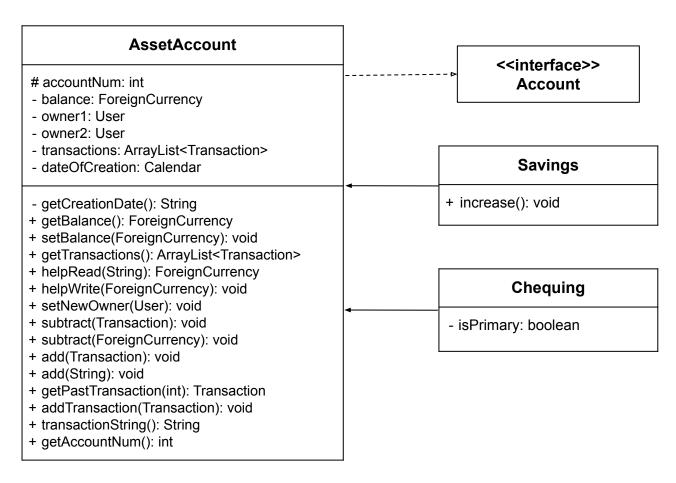
<<interface>>
Account

LineOfCredit

CashBackCard

+ increase(): void

package phase2.Accounts



package phase2.Data

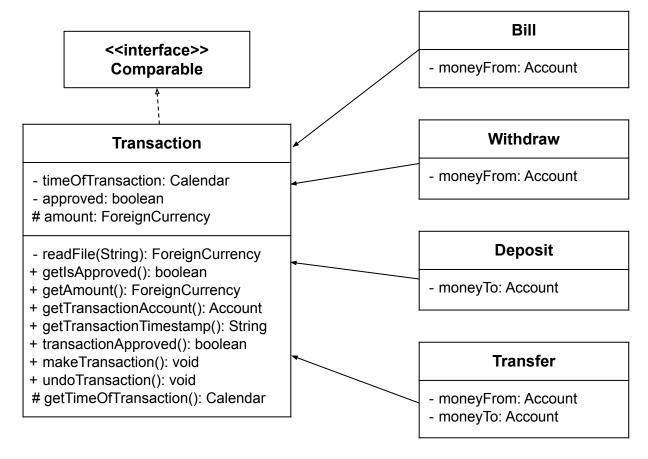
DataSaver

- writeATMData(): void
- writeAllUsers(): void
- writeUserData(User): void
- appendName(User): void
- writeSavingAccount(FileWriter, Savings): void
- writeAllSavings(FileWriter, User): void
- writeChequingAccount(FileWriter, Chequing): void
- writeAllChequing(FileWriter, User): void
- writeCreditCardAccount(FileWriter, CreditCard): void
- writeAllCreditCard(FileWriter, User): void
- writeLineOfCreditAccount(FileWriter, LineOfCredit): void
- writeAllLineOfCredit(FileWriter, User): void
- writeCashBackCardAccount(FileWriter, CashBackCard): void
- writeAllCashBackCard(FileWriter, User): void
- writeAllAccountRequests(): void
- writeAllUndoRequests(): void
- writeAllUserRequests(): void

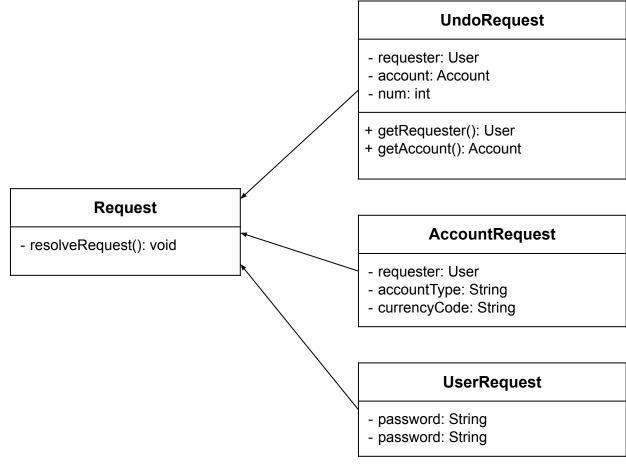
DataReader

- types: String[]
- readAllUserData(): void
- readUserData(String): void
- initializeUser(String): User
- addAccounts(String, User, int, BufferedReader):
 void
- addAllAccounts(String, User, BufferedReader):
 void
- initializeAccount(String, User, String): Account
- getCorrectCalendar(String[]): GregorianCalendar
- addAllTransactions(String, Account): void
- readTransaction(String[], Account): void
- readTransfer(String[], Account): void
- readOtherTransaction(String[], Account): void

package phase2.Transactions



package phase2.Request



package phase2.People

User

- username: Stringpassword: String
- accounts: AccountManager
- getTransactions(): ArrayList<Transaction>
- + requestAccount(String, String): void
- + requestUndo(Account, int): void
- + getAccountManager(): AccountManager
- + getUsername(): String
- + getPassword(): String
- + setPassword(String): void
- + makeTransaction(Transaction): void
- # getTransaction(int): Transaction
- + balanceSummary(): String
- + getAccountList(String): ArrayList<Account>
- + addAccountOwner(String, int): void
- # getAccount(int): Account

BankTeller

- getNumberOfRequests(): int
- getIndex(String): int
- deleteAlerts(): void
- + getRequestSummary(): String
- + addRequest(UndoRequest): void
- + ignoreRequest(int): void
- + completeRequest(int): void
- + restockCashMachine(): void
- # getRequest(int): UndoRequest

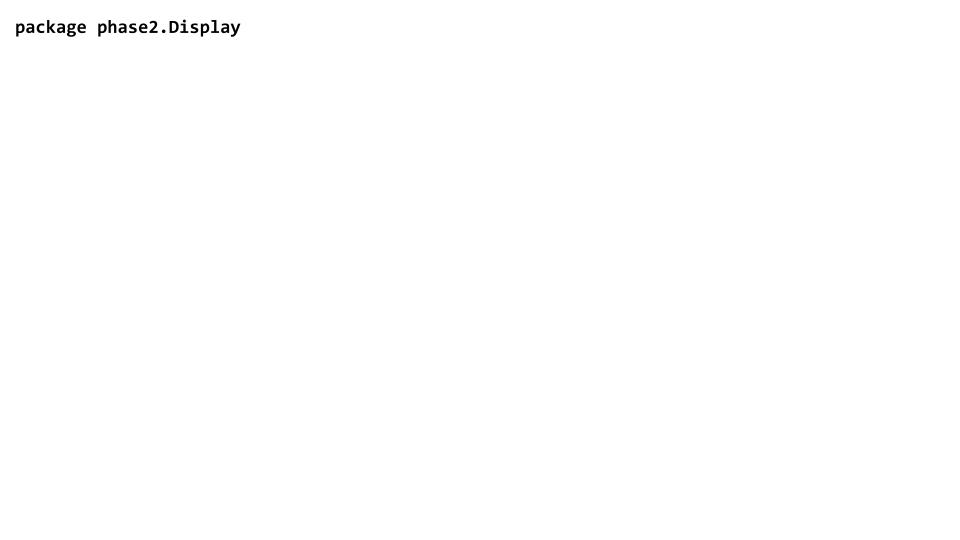
package phase2.People

UserManager

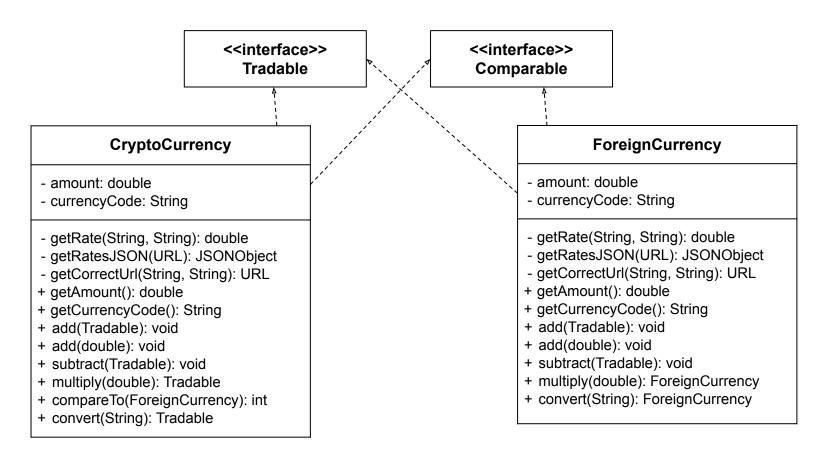
- + accountNum: int
- getCreationDate(): String
- + authenticateUsername(String): boolean
- + isBankManager(String): boolean
- + getNextAccountNum(): int
- + authenticatePassword(String, User): boolean
- + getUser(String): User
- + getUserAccount(int): Account
- + addUser(String, String): void
- + updateInterest(): void

BankManager

- username: Stringpassword: String
- + userRequests: ArrayList<UserRequest>
- + accountRequests: ArrayList<AccountRequest>
- + getUsername(): String
- + getPassword(): String
- + setDate(int, int, int, int, int, int): void
- + getRequestList(): ArrayList<Request>
- + getRequest(int): Request
- + getNumberOfRequests(): int
- getIndex(String): int
- deleteAlerts(): void
- + getRequestSummary(): String
- + addRequest(UndoRequest): void
- + ignoreRequest(int): void
- + completeRequest(int): void
- + restockCashMachine(): void



package phase2.Tradable



package phase2

CashMachine

- bills: int[4]
- getNumBills(int): int
- getDenomination(int): int
- decreaseBills(int, int): void
- writeAlerts(): void
- numSubtract(int, int): int
- + increaseBills(int, int): void
- + withdrawBills(double): void
- · Withdraw Dills (double). Ve
- + totalBalance(): int

ATMTime

- yearFactor: int
- monthFactor: int
- dayFactor: int
- hourFactor: int
- minuteFactor: int
- secondFactor: int
- getCurrentDay(): int
- + getCurrentTime(): Calendar
- + isFirstOfMonth(): boolean
- + setDate(int, int, int, int, int, int): void