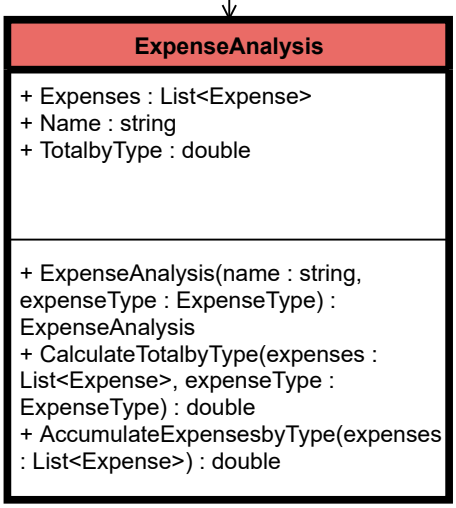
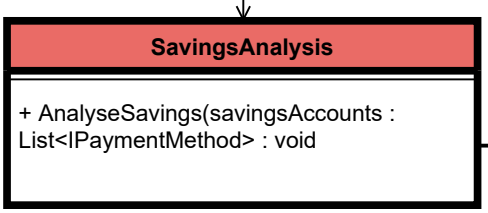
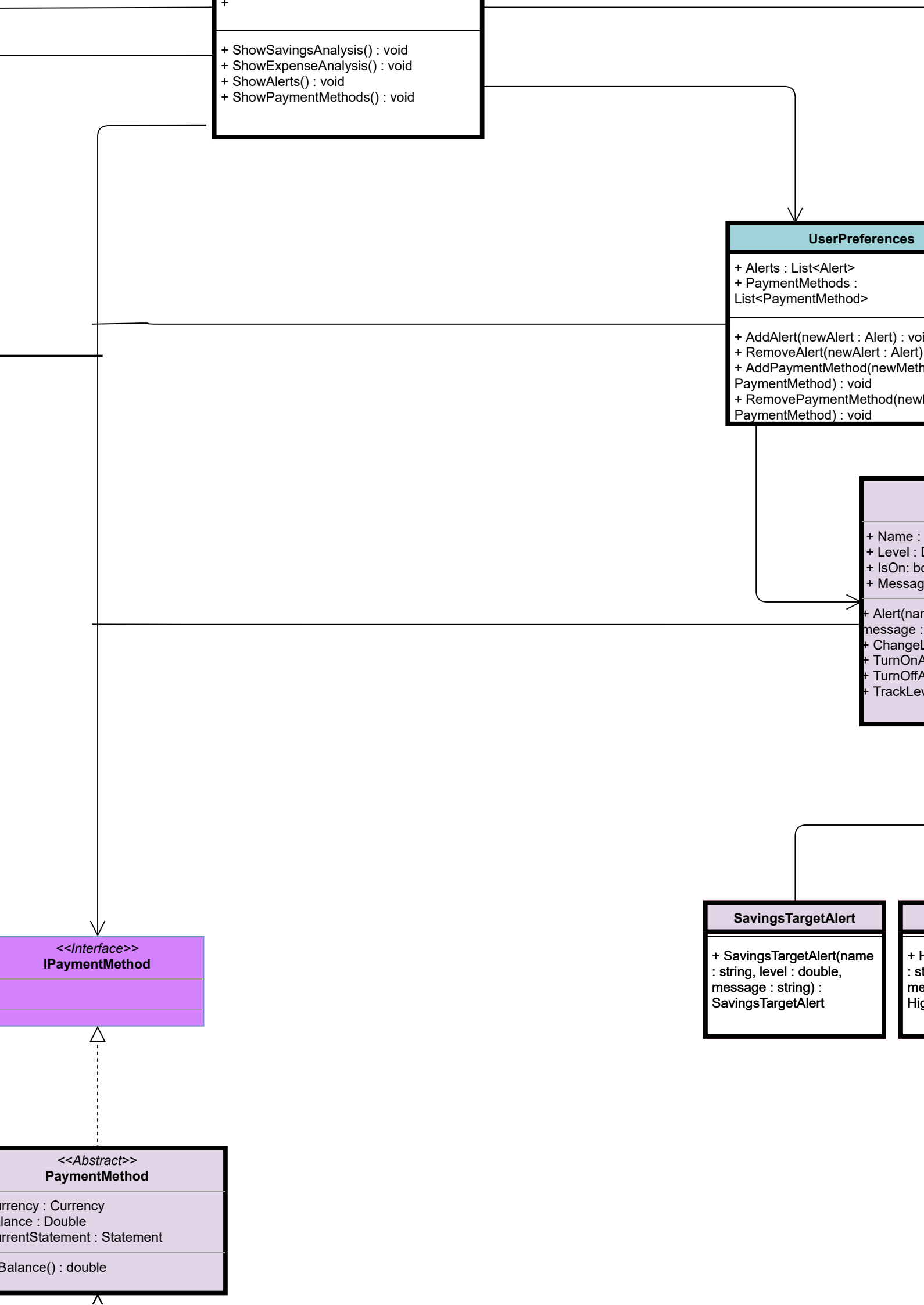
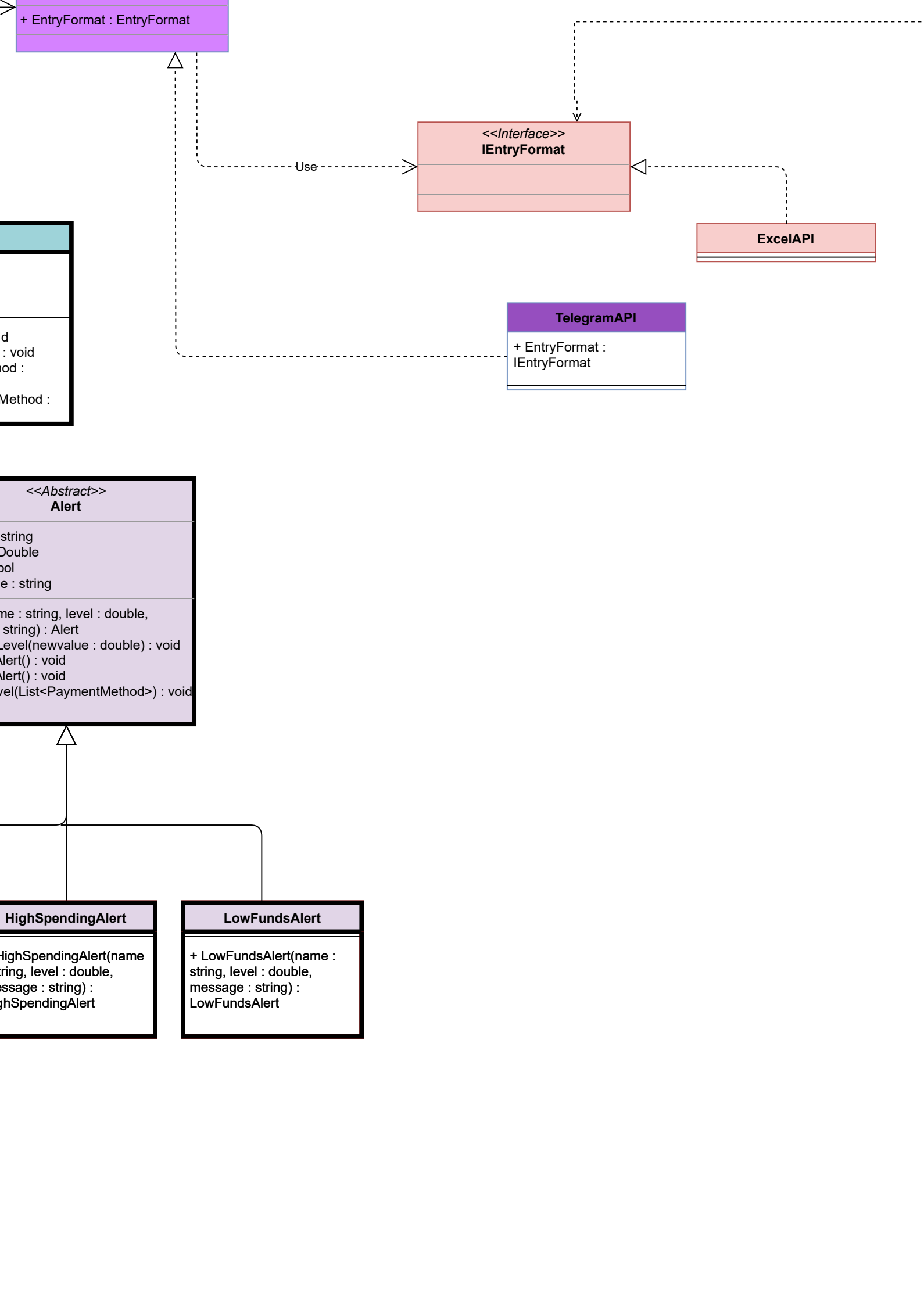
UserInterface

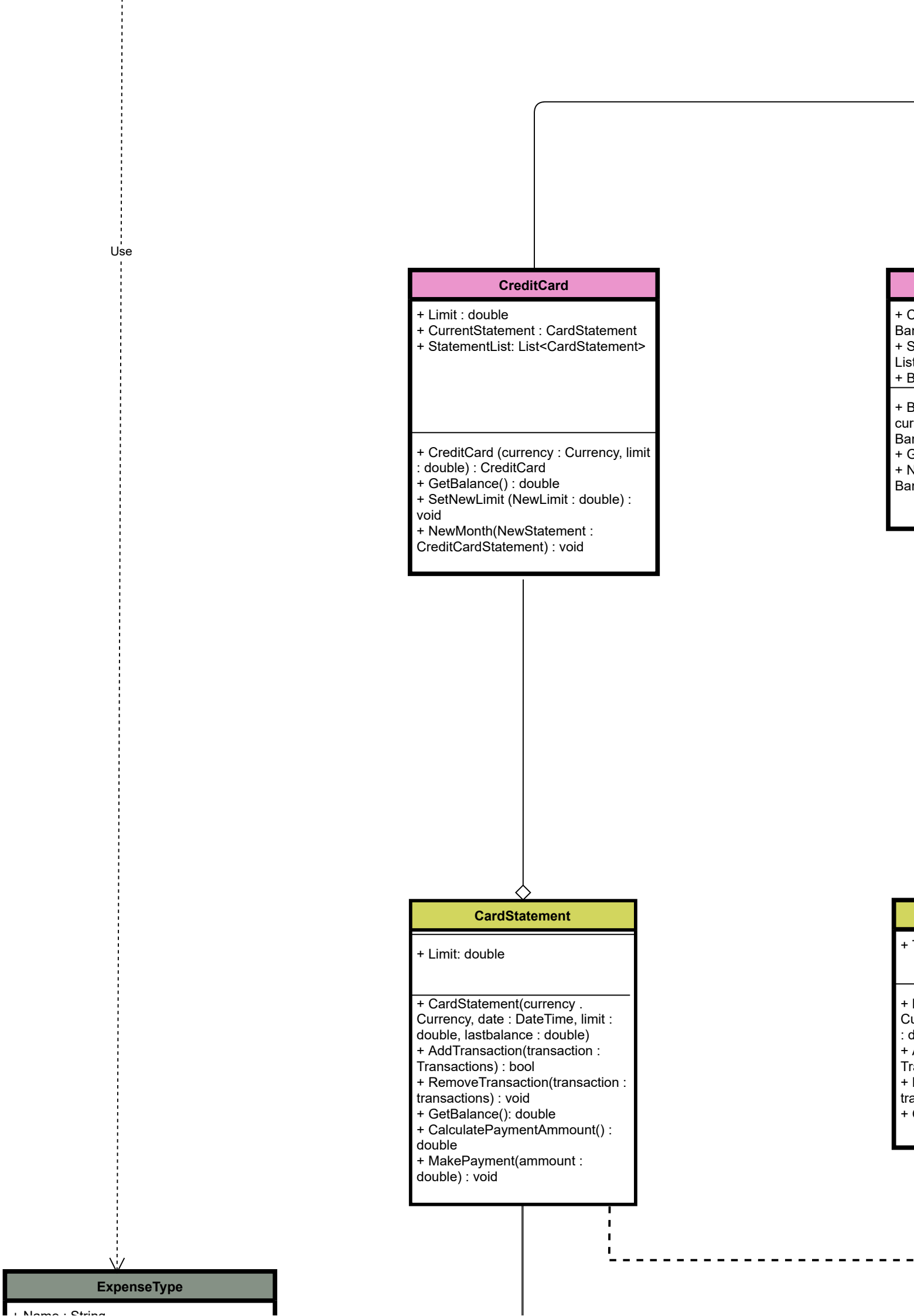
<<Interface>>
IExitFormat

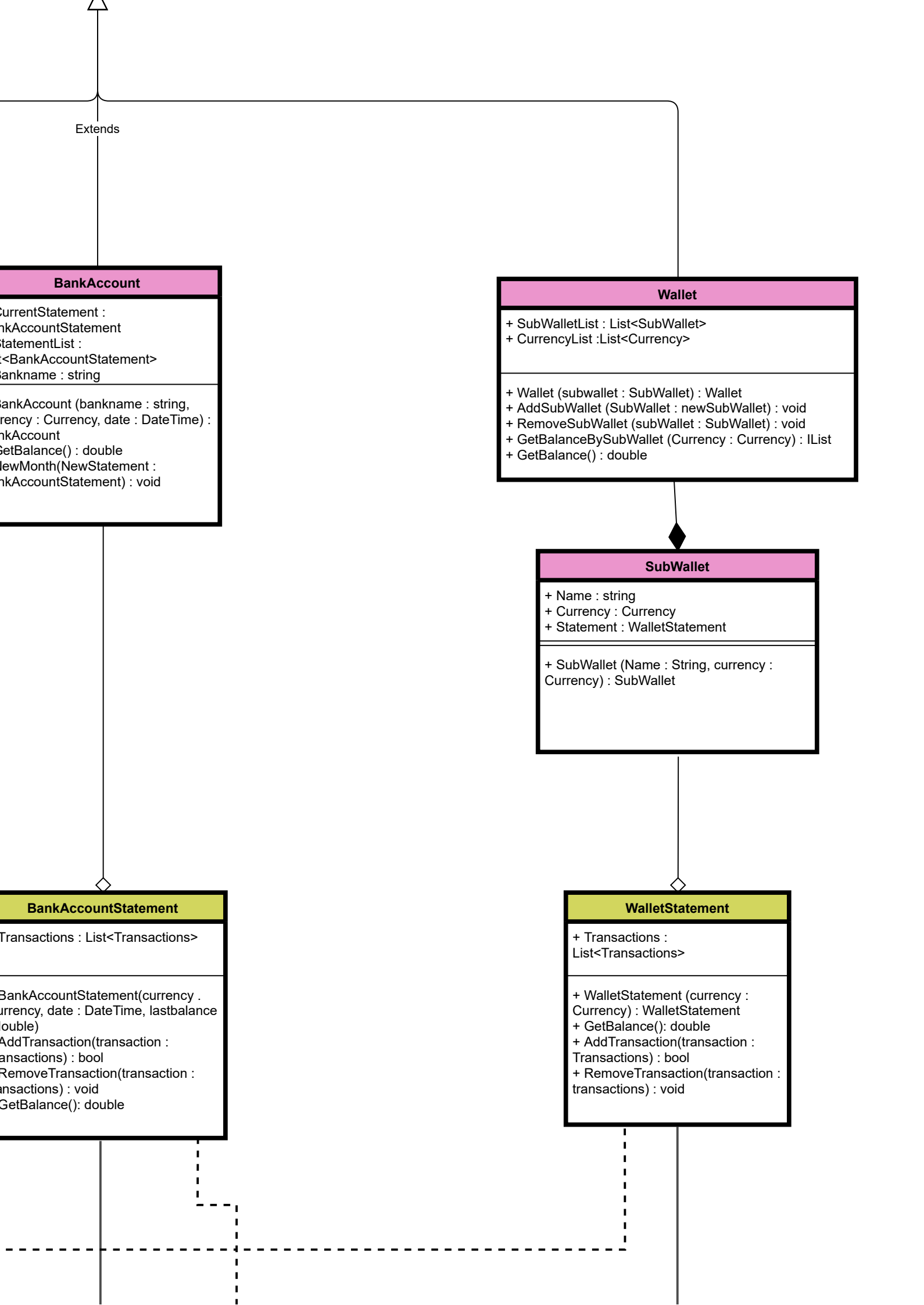

ExpenseAnalysis
+ Expenses : List<Expense>
+ Name : string
+ TotalbyType : double
+ ExpenseAnalysis(name : string, expenseType : ExpenseType) : ExpenseAnalysis
+ CalculateTotalbyType(expenses : List<Expense>, expenseType : ExpenseType) : double
+ AccumulateExpensesbyType(expenses : List<Expense>) : double

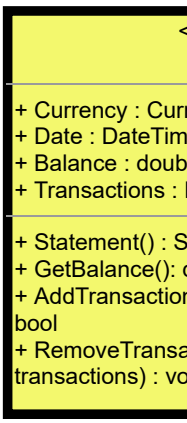

SavingsAnalysis
+ AnalyseSavings(savingsAccounts : List<IPaymentMethod> : void

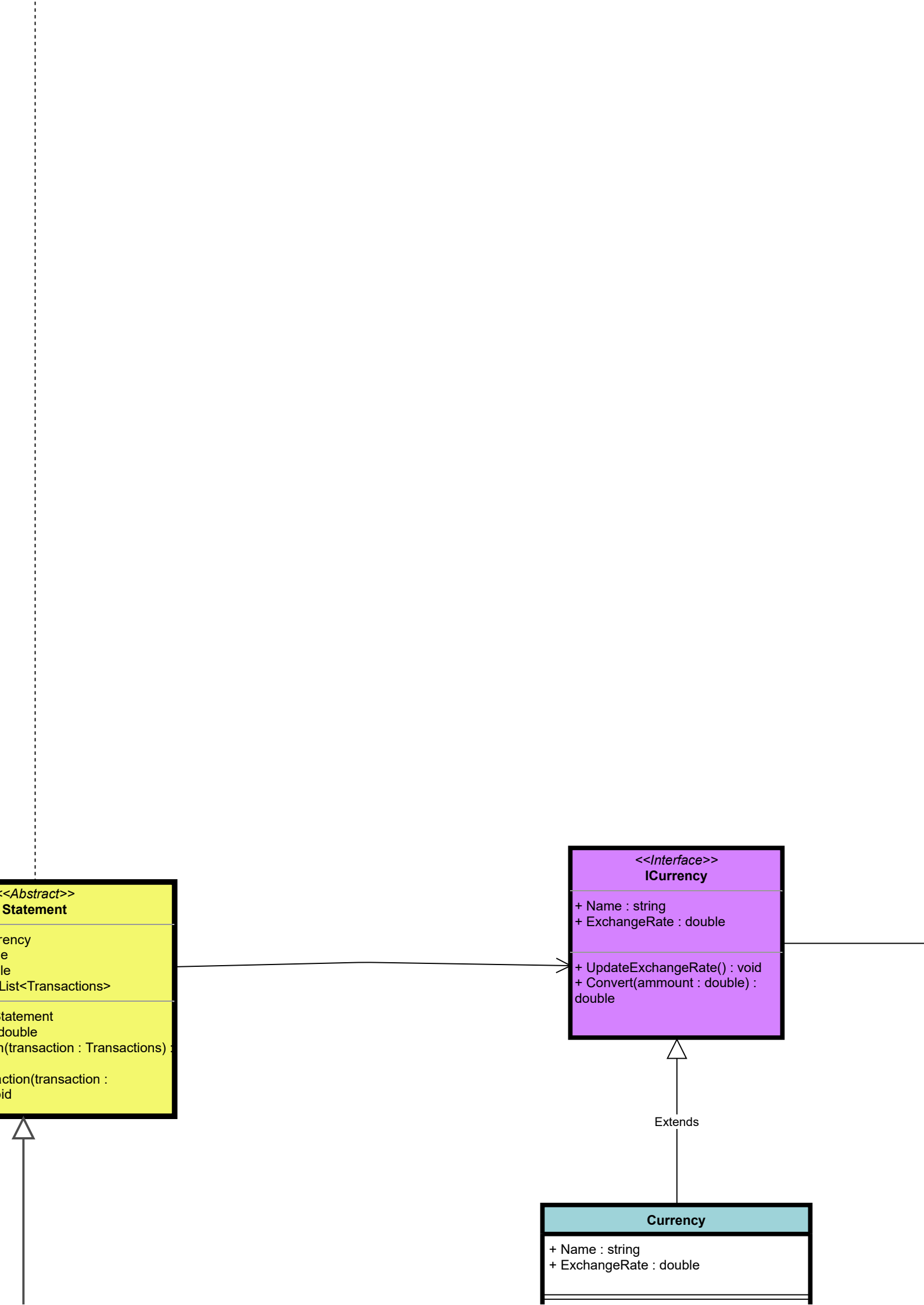

+
+ ShowSavingsAnalysis() : void
+ ShowExpenseAnalysis() : void
+ ShowAlerts() : void
+ ShowPaymentMethods() : void
UserPreferences
+ Alerts : List<Alert>
+ PaymentMethods :
List<PaymentMethod>
+ AddAlert(newAlert : Alert) : voi
+ RemoveAlert(newAlert : Alert)
+ AddPaymentMethod(newMeth
PaymentMethod) : void
+ RemovePaymentMethod(newl
PaymentMethod) : void
+ Name :
+ Level : D
+ IsOn: bo
+ Messag
+ Alert(nar
message :
+ ChangeL
+ TurnOnA
+ TurnOffA
+ TrackLev
SavingsTargetAlert
+ SavingsTargetAlert(name
: string, level : double,
message : string) :
SavingsTargetAlert
+ H
: st
me
Hig
<<Interface>>
IPaymentMethod
<<Abstract>>
PaymentMethod
rrency : Currency
lance : Double
rrentStatement : Statement
Balance() : double

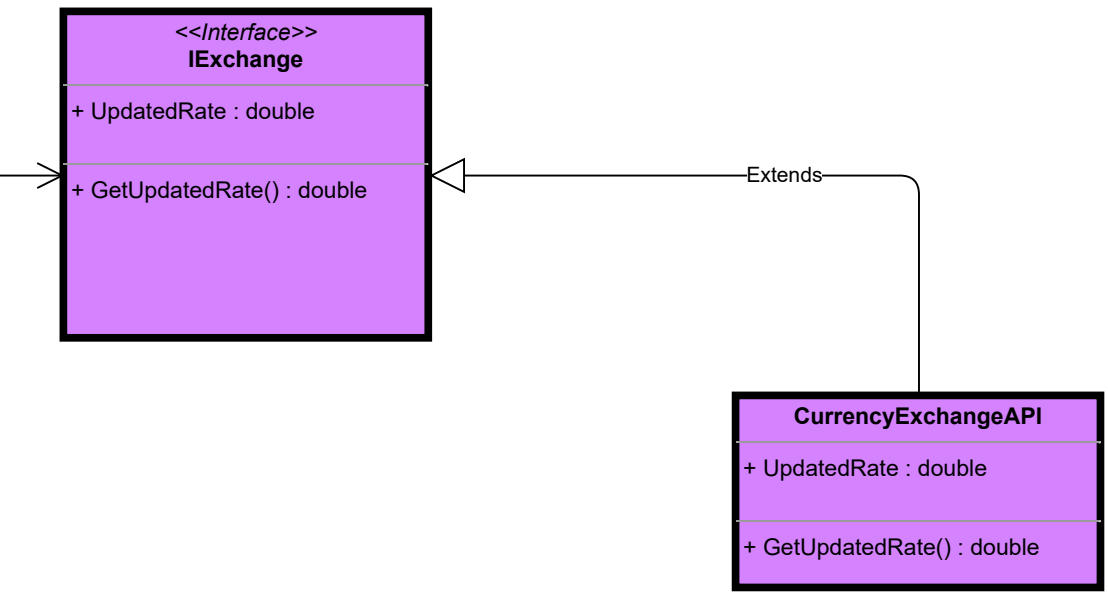

+ EntryFormat : EntryFormat
<<Interface>>
IEntryFormat
Use
ExcelAPI
TelegramAPI
+ EntryFormat :
IEntryFormat
<<Abstract>>
Alert
HighSpendingAlert
LowFundsAlert
+ LowFundsAlert(name :
string, level : double,
message : string) :
LowFundsAlert

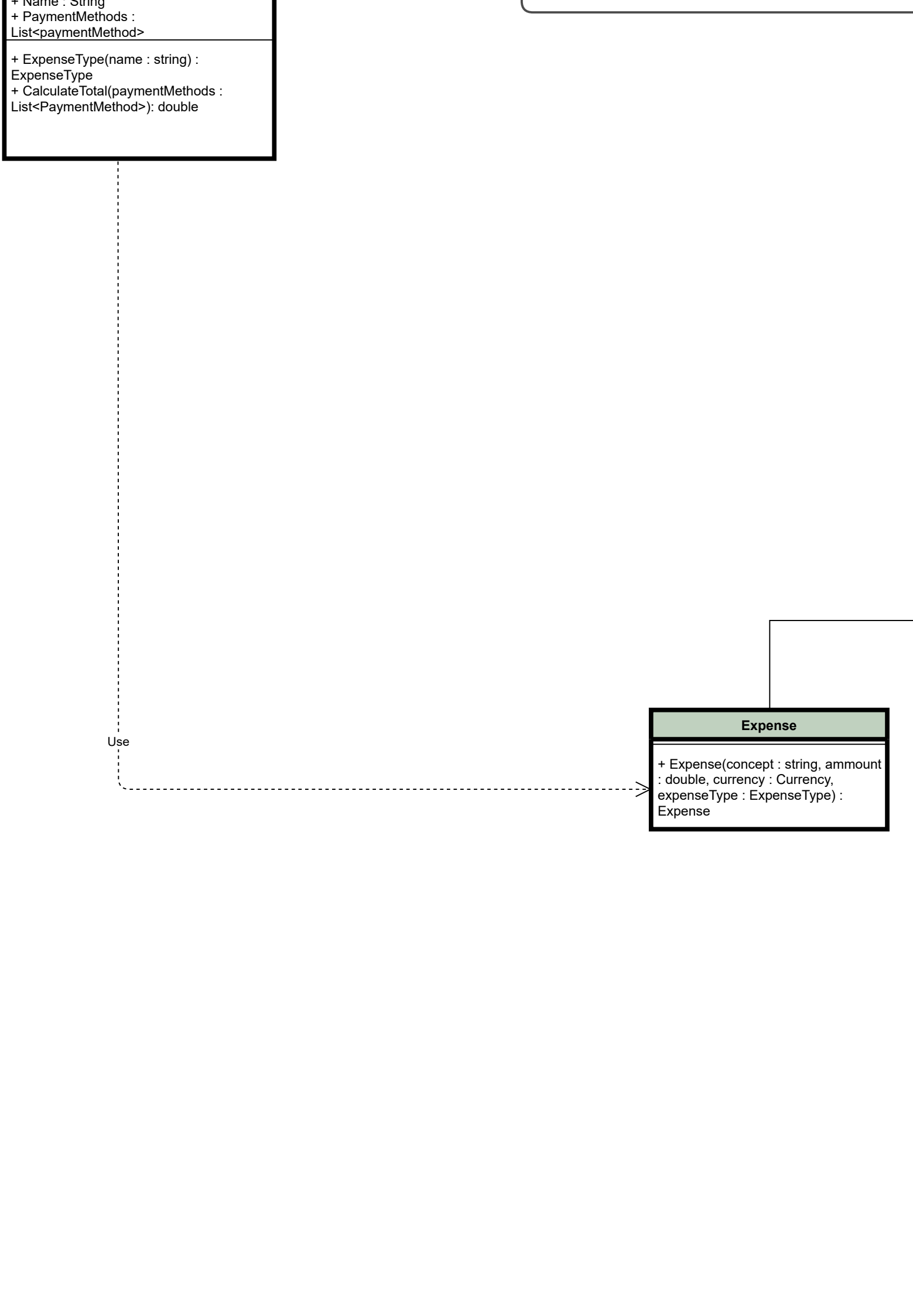

Use
CreditCard
+ Limit : double
+ CurrentStatement : CardStatement
+ StatementList: List<CardStatement>
+ CreditCard (currency : Currency, limit : double) : CreditCard
+ GetBalance() : double
+ SetNewLimit (NewLimit : double) : void
+ NewMonth(NewStatement : CreditCardStatement) : void
CardStatement
+ Limit: double
+ CardStatement(currency . Currency, date : DateTime, limit : double, lastbalance : double)
+ AddTransaction(transaction : Transactions) : bool
+ RemoveTransaction(transaction : transactions) : void
+ GetBalance(): double
+ CalculatePaymentAmmount() : double
+ MakePayment(ammount : double) : void
ExpenseType

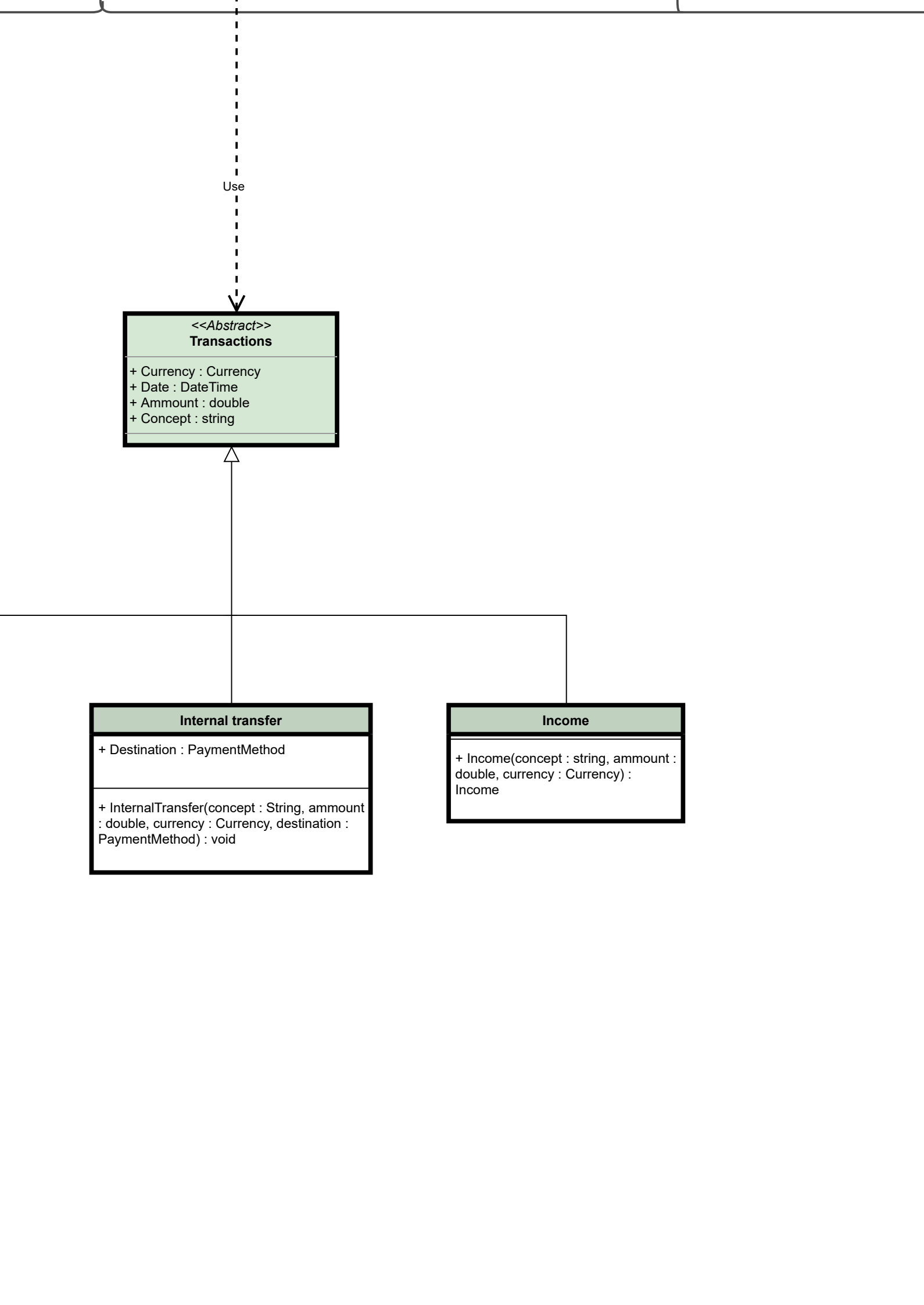

Extends
BankAccount
urrentStatement :
nkAccountStatement
tatementList :
t<BankAccountStatement>
ankname : string
ankAccount (bankname : string,
rency : Currency, date : DateTime) :
nkAccount
etBalance() : double
lewMonth(NewStatement :
nkAccountStatement) : void
Wallet
+ SubWalletList : List<SubWallet>
+ CurrencyList :List<Currency>
+ Wallet (subwallet : SubWallet) : Wallet
+ AddSubWallet (SubWallet : newSubWallet) : void
+ RemoveSubWallet (subWallet : SubWallet) : void
+ GetBalanceBySubWallet (Currency : Currency) : IList
+ GetBalance() : double
SubWallet
+ Name : string
+ Currency : Currency
+ Statement : WalletStatement
+ SubWallet (Name : String, currency :
Currency) : SubWallet
BankAccountStatement
Transactions : List<Transactions>
BankAccountStatement(currency .
urrency, date : DateTime, lastbalance
louble)
AddTransaction(transaction :
ansactions) : bool
RemoveTransaction(transaction :
nsactions) : void
GetBalance(): double
WalletStatement
+ Transactions :
List<Transactions>
+ WalletStatement (currency :
Currency) : WalletStatement
+ GetBalance(): double
+ AddTransaction(transaction :
Transactions) : bool
+ RemoveTransaction(transaction :
transactions) : void

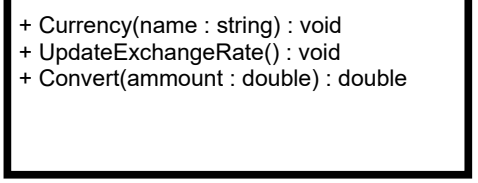

+ Currency : Curr
+ Date : DateTim
+ Balance : doub
+ Transactions :
+ Statement() : S
+ GetBalance(): d
+ AddTransaction
bool
+ RemoveTransa
transactions) : vo


<<Abstract>>
Statement
rency
e
le
List<Transactions>
tatement
double
n(transaction : Transactions) :
ction(transaction :
id
<<Interface>>
ICurrency
+ Name : string
+ ExchangeRate : double
+ UpdateExchangeRate() : void
+ Convert(ammount : double) : double
Extends
Currency
+ Name : string
+ ExchangeRate : double


<<Interface>>
IExchange
+ UpdatedRate : double
+ GetUpdatedRate() : double
Extends
CurrencyExchangeAPI
+ UpdatedRate : double
+ GetUpdatedRate() : double


+ Name : String
+ PaymentMethods :
List<paymentMethod>
+ ExpenseType(name : string) :
ExpenseType
+ CalculateTotal(paymentMethods :
List<PaymentMethod>): double
Use
Expense
+ Expense(concept : string, ammount
: double, currency : Currency,
expenseType : ExpenseType) :
Expense


Use
<<Abstract>>
Transactions
+ Currency : Currency
+ Date : DateTime
+ Ammount : double
+ Concept : string
Internal transfer
+ Destination : PaymentMethod
+ InternalTransfer(concept : String, ammount : double, currency : Currency, destination : PaymentMethod) : void
Income
+ Income(concept : string, ammount : double, currency : Currency) : Income


+ Currency(name : string) : void
+ UpdateExchangeRate() : void
+ Convert(ammount : double) : double