**Bank Account Program**

­­

StopWatch’s object to check elapsed time

Contains time interval for increment

Creates SavingAccount object with specified id and balance.

Returns balance.

Withdraws amount from balance (withdraw limit is 20%).

Deposits amount from balance.

Increments by 5% of current balance every after 30 second

**SavingAccount**

- timer: StopWatch

- timeInterval: double

SavingAccount (double balance)

+ getBalance (): double

+ withdraw (double amount): void

+ deposit (double amount): void

- savingAccountInterest (): void

**CheckingAccount**

CheckingAccount (double balance)

+ withdraw (double amount): void

Creates CheckingAccount object with specified id and balance.

Withdraws amount from balance (overdraft of 30% is possible).

\* Checking Account have overdraft feature.

\* Saving Account has some different features, when we create account with certain balance, balance is increased by 5% of current balance after every 30 seconds until program runs. To increment balance after 30 seconds duration a **StopWatch** object is used.

The id of account.

The Balance in account.

Array which holds all type of accounts.

Counter of created accounts.

Creates Account object with specified id and name.

Returns balance.

Withdraws amount from balance.

Deposits amount from balance.

Return boolean but its actual task is to print accounts’ menu in which all types of accounts are shown and creates user desired account.

Add newly created account in **accounts** array.

Returns **accounts** array.

**Account**

# id: int: {read only}

# balance: double

- accounts: Account []

- numberOfAccounts: int

Account (double balance)

+ getBalance (): double

+ withdraw (double amount): void

+ deposit (double amount): void

+ accountsMenu (): boolean

+ addAccount (Account a): void

+ getAccountsArray (): Account []