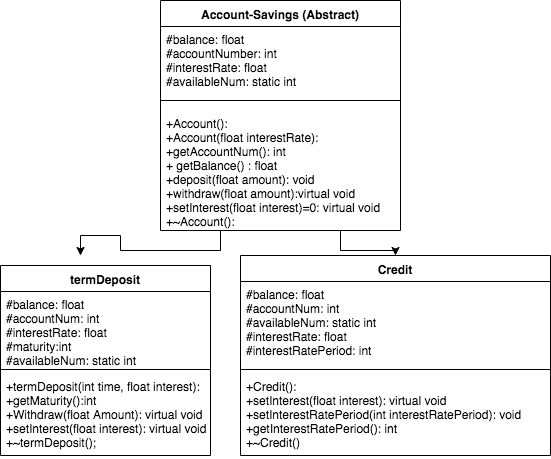
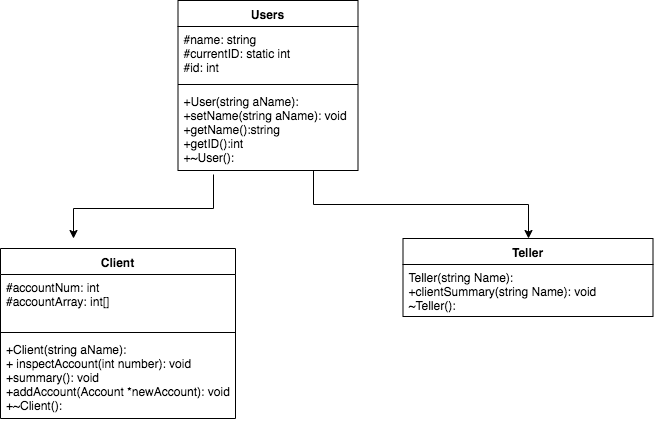
****

**VARIABLES:**

Account:

**Balance**, Stores the balance of this account.

**accountNumber**, Stores the number that belongs to this account

**interestRate**, stores the current interest rate.

**availableNum**, Stores the current available account number

termDepost:

**balance**, Stores the balance of this term deposit account

**accountNum**, Stores the number that belongs to this term deposit account

**interestRate**, Stores the interest rate belonging to this term deposit

**maturity**, Stores the amount of time remaining until the balance can be withdrawn.

**avaliableNum**, Stores the available term deposit account number which is on the stack.

Credit:

**Balance**, Stores the balance of this Credit account.

**accountNum**, Stores the number that belongs to this Credit Account.

**availableNum**, Stores the current available Credit account number which is on the stack

**interestRate**, Stores the interest rate belonging to this credit Account.

**interestRatePeriod**, Stores the period in which the current interest rate stands.

Users:

**Name**: Stores the name of the user

**CurrentID**: Stores the current available ID number of the user which is on the stack.

**Id**: Stores the ID number of the user.

Client:

**accountNum**, Stores the account number of the client

**accountArray**, Stores a dyanmic array of all the accounts belonging to the Client

**METHODS:**

Account:

**Account**(), Constructs an account object with no specifics, but it has an account number stored in **accountNumber** and it has a balance of 0 sorted in **balance**. It is a very general account object

**Account**(float **interestRate**), Constructs an account object with the given parameter which will be stored in the **Interest** Variable and it will start with a **Balance** of 0. This will also be given an individual account number specific to its type that will be stored in **accountNumber**.

**getAccountNum**(), This member function will return the account number specific to an object that is storied in the **accountNumber** variable.

**getBalance**(), This member function will return the balance specific to an object that is stored in the **balance** variable.

**Deposit**(float **amount**) This member function will take the given amount and check if the variable amount is not a negative. If it is negative, then it will display error message. If not then it will take that value and add it to the variable **balance**.

**Withdraw**(float **amount**). This member function is a virtual function. In this class it will take the given amount and check if it is higher than the number stored in **balance**. If it is then it will display error message. If not, then it will subtract number stored in amount from the variable **balance**.

**setInterest**(float **interest**) this member function is a pure virtual function and will not be defined in this class.

~**Account**(), This member function will delete the account object and free all variables.

termDepost:

**termDeposit**(int **time**, float **interest**), This member function will construct a **termDeposit** object that inherits from **Account** with the given parameters. This object will contain an account number of the available account number from **availableNum** stored in **accountNum** variable and it will also have a variable **maturity** which will contain the value assigned to **time** and an **interestRate** variable which will contain the value assigned to **interest** in the parameter.

**Withdraw**(float **amount**), this member function will check to see if the variable **maturity** is at 0, if it isn’t then it will display a message to let the user know they cannot withdraw. If it is 0, then it will subtract amount from the **balance** variable.

**setInterest**(float interest), This member function will update the current set interest rate which is stored in the variable **interestRate**. This shows polymorphism as this member function works for this subclass and not the parent.

~**termDepost**(), This member function will delete the **termDepost** object and free all variables.

Credit:

Credit(), This member function will construct a Credit object that inherits from Account. This has an account number which is assigned from the **availableNum** variable and is stored in **accountNum**. It also has an interest rate stored in **interestRate** which is initialised to 0 and an **interestRatePeriod** which is also initialised at 0.

**setInterest**(float **interest**), This member function will update the current set interest rate which is stored in the variable **interestRate**. This shows polymorphism as this member function works for this subclass and not the parent.

**setInterestRatePeriod**(int **period**), This member function will update the current interest rate period which is stored in **interestRatePeriod.**

**getInterestRatePeriod**(), This member function will return the current interest rate period stored in **interestRatePeriod.**

~**Credit**(), This member function will delete the **Credit** object and free all variables.

Users:

**User**(string **aName**), This member function will create an object user with the given parameter **aName** which will be stored in variable **Name**. It will have an ID stored in **id** which is will receive from the **currentID** variable.

**setName**(string **aName**) This member function allows the user to change the existing name of the user which will be taken by the parameter **aName** and stored in **Name**.

**getName**(), This member function will return the **Name** variable which contains the users name.

**getID**(), This member function will return the users ID stored in the **id** variable.

~**User**(), This member function will delete the **User** object and free all variables.

Client:

**Client**(string **aName**), This member function creates a Client object which will store its name from the given parameter in the variable **Name** from its parent class **User**. It will then assign it an **accountNum** of 0.

**inspectAccount**(int **number**), This member function takes a given account number and finds its details on the dynamic array **accountArray.** Then it will output the account number stored in **accountNum**, its **balance** and the **interestRate**.

**Summary**(), This member function will output all accounts owned by a Client and its balance. It will use a for loop to output its account number from the **accountArray** which will store the name of an account object. Using that name it will use its member function **getAccountNum**() to get the account number. The object accounts member function **getBalance**() will be used to get the associated balance. Which will be found using its **accountNum**.

**addAccount**(\*Account **newAccount**), This member function will add an object account to the **accountArray**.

~**Client**(), This member function will delete the **Client** object and free all variables.

Teller:

**Teller**(string **Name**), This member function construct an object of type teller which inherits from the **user** class. Its parameter takes a string as a variable which is the name of the user and creates a teller.

**clientSummary**(Client\* **CArray**[], int **size**, string **client**), This member function takes in the client array (**CArray**) which is an array of all clients and the **size** of that array with the clients **name**. Using a for loop, this locates the the clients name, whose index is saved to the variable **point**. Using this variable, another for loop is used to act as the amount of accounts for the person and using that loop it outputs the Account number stored in **accountNum** and the **balance** to the user.

~**Teller**(), This member function will delete the **Teller** object and free all variables.