Logo

Description automatically generated

**22/SU-COP-2805C-72035 Java Programming**

**Exercise 11.8**

Document Version: 0.1

Version Date: June 10, 2022

Created By: Johnathan Webster

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Rationale |
| 0.1 | 2022 June 10 | Johnathan Webster | First Draft |

# Document Purpose

The purpose of this document is to define the updated New\_Account() class.

# Technical Specifications

## Purpose of Technical Implementation

The purpose of this implementation is to update the previous account class to include the Transaction() class and update the properties of the Account() class to include new fields.

## Technical Implementation Components

The class implementation allows the user to create an account with new criteria, if specified.

The first update is the properties. The class now includes a name files and transactions field of type Arraylist.

//*Properties*

*private* int id;

*private* double balance;

*private* double annualInterestRate;

*private* LocalDate dateCreated;

*private* String name;

*private* ArrayList<Transaction> transactions;

The second update is the constructors. Now if the program creates an instance of and account and provide no arguments the instance will have defaulted values inserted.

*Account*(){

    name = "";

    id = 0;

    balance = 0;

    annualInterestRate = 0;

    transactions = *new* ArrayList<Transaction>();

*setDateCreated*();

}

If an account ID and Balance are provided the object will be instantiated with the specified values.

*Account*(int id, double balance){

    name = "";

    this.*id* = 0;

    this.*balance* = 0;

    annualInterestRate = 0;

    transactions = *new* ArrayList<Transaction>();

*setDateCreated*();

}

Lastly if a name is provided with the ID and balance the object will include this also.

*Account*(String name, int id, double balance){

    this.*name* = name;

    this.*id* = id;

    this.*balance* = balance;

    transactions = *new* ArrayList<Transaction>();

*setDateCreated*();

}

The third update is the addition of the Transactions class. The transaction class will provide a stamp with the provided information below on each deposit or withdrawal.

*private* java.util.Date date;

*private* char transactionType;

*private* double ammount;

*private* double balance;

*private* String description;

Each constructor for the transactions will update the above criteria each time a transaction is made.

*Transaction*(char transactionType, double ammount, double balance, String description){

        this.*transactionType* = transactionType;

        this.*ammount* = ammount;

        this.*balance* = balance;

        this.*description* = description;

        this.*date* = *new* java.*util*.*Date*();

    }

*Transaction*(double ammount, double balance, String description){

        this.*ammount* = ammount;

        this.*balance* = balance;

        this.*description* = description;

        this.*date* = *new* java.*util*.*Date*();

    }

*Transaction*(double ammount, double balance){

        this.*ammount* = ammount;

        this.*balance* = balance;

        this.*date* = *new* java.*util*.*Date*();

    }

\*with respective accessors and modifiers

## New Account Class Implementation

The implementation of this class allows the user to create a new account. Now, the user can enter a name and anytime a transaction is made the transaction information is stored in an arraylist and can be called to provide all transactions for the account.