

**<22/SP-COP-2800-72035> Java Advanced**

**<Assignment 11-08>**

Document Version: 0.1

Version Date: July 1, 2022

Created By: David Duron

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Rationale |
| 0.1 | 2022 JUL 1 | David Duron | Submit Assignment |

# Document Purpose

The purpose of this document is to define the Transaction class and discuss how to implement and use it.

# Technical Specifications

## Purpose of Technical Implementation

The purpose of the Transaction is store information when a Transaction is created. The following information will be attributed to the newly created object: type, amount, balance, description. The Transaction class has the ability to view the transaction date, type, amount, balance, description. The class can edit the type, amount, balance, description. The Transaction class is dependent on one Java APIs: java.util.Date..

## Technical Implementation Components

I created the Transaction class. The Transaction class has four properties (id, balance, annualInterestRate, and dateCreated) and multiple methods (setBalance, setInterestRate, getId, getBalance, getAnnualInterestRate, getMonthlyInterestRate, getMonthlyInterest, getDateCreated, setId, deposit, withdraw, setAnnualInterestRate).

**Properties**

1. Date: current timestamp
2. type: w = withdraw, d = deposit
3. amount: the amount to add or subtract from existing balance
4. balance: the new balance after the transaction
5. description: a string that tells what kind of transaction was performed

**Methods**

1. getDate(): displays date of transaction
2. getType(): display type of transaction
3. getAmount(): displays amount of transaction
4. getBalance(): display balance of transaction
5. getDescription: display description of transaction
6. setType(char type): sets type of transaction
7. setAmount(double amount) set amount of transaction
8. setBalance(double balance) set balance of transaction
9. setDescription(String description) set description of transaction

**Constructors**

The developer can create an instance of the Transaction class one way, it can be implemented in the deposit and withdraw methods of the Account class.

1. Transaction Transaction\_example1 = new Transaction(char type, double amount, double balance, String description);

## Technical Implementation Pseudocode

Create instance of the Transaction class using the first type of constructor

Use appropriate method to display properties

Use appropriate method to set property values

End