

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

**<Assignment 11-03>**

Document Version: 0.1

Version Date: June 28, 2022

Created By: David Duron

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Rationale |
| 0.1 | 2022 JUN 28 | David Duron | Submit Assignment |

# Document Purpose

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

# Technical Specifications

## Purpose of Technical Implementation

The purpose of the Account is store information when an account is created. The following information will be attributed to the newly created object: id, balance, annual interest rate. The account class has the ability to view the id, balance, creation date, interest rate (monthly, annual), and the monthly interest payment. The Account class is dependent on three Java APIs: java.time.Instant, java.util.Date, and java.text.DecimalFormat.

## Technical Implementation Components

I created the Account class. The Account 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. id is the id of the account that was created. This will be auto-incremented when connected to a database.

2. balance is the account balance

3. annualInterestRate is how much the account will over time

4. dateCreated is the date the account was created. This is created used Java 8’s Instant class which stores a unix timestamp once the object is created.

**Methods**

1. Private setBalance,
2. Private setInterestRate,
3. getId displays the user’s id,
4. getBalance displays the account balance,
5. getAnnualInterestRate displays the annual interest rate,
6. getMonthlyInterestRate displays the calculated monthly interest rate,
7. getMonthlyInterest displays the calculated monthly interest rate in a readable format,
8. getDateCreated displays the unix timestamp in a readable format,
9. setId used to set the account id,
10. deposit sets the account balance to the current balance plus an additional amount provided by the user,
11. withdraw sets the account balance to the current balance minus an additional amount provided by the user,
12. setAnnualInterestRate = sets the annual interest rate with a provided amount

**Constructors**

The developer can create an instance of the Account class three different ways. The first way is recommended for new stocks or if the developer wants to save to the database and obtain an id to best used later. The second way is ideal for accounts requiring a minimum account balance. The third way is useful when the developer has already has obtained an id, a balance, and knows the annual interest rate.

1. Account account\_example1 = new Account ();

2. Account account\_example2 = new Account(id, balance);

3. Account account\_example3 = new Account(id, balance, annualInterestRate);

## Technical Implementation Pseudocode

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

Set id

Set balance

Set annual interest rate

Or

Create instance of the Account class using the second type of constructor

Set annual interest rate

Or

Create instance of the Account class using the third type of constructor

Then

Display balance

Display id

Display creation date

Display annual interest rate

Display monthly interest rate

Display monthly interest

Deposit into the account

Withdraw from the account

End