

**22/SP-COP-2805C-61163 Java Advanced**

**Assignment 9.7**

Document Version: 0.1

Version Date: 04/12/2022

Created By: Tristan Rogers

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Rationale |
| 0.1 | 2022 APR 12 | Tristan Rogers | First Draft |

# Document Purpose

The purpose of this document is to outline the process behind the Account class and discuss the parameters and functions involved.

# Technical Specifications

## Purpose of Technical Implementation

The application creates a new account using specific parameters. It enables the user to deposit and withdraw funds from the account, while also showing the monthly interest on the account and the date that the account was created.

## Technical Implementation Components

* The program uses one class named Account. Within this Account class, there are accessors and mutators for each of the fields, which are account ID, account balance, Annual interest rate, and monthly interest/ interest rate. Two methods named withdraw and deposit, allow the user to respectively withdraw and deposit funds to the account. The method named getDateCreated allows the user to see when the account was created. The main method is what runs the test to ensure that the code is working as intended.

## Technical Implementation Pseudocode

*Initialize variables with a default value of 0*

*create account with id / balance values and current date*

*get interest rate based on balance of account*

*allow withdraw and/or deposit based on user intentions*

*display results to user*

## Account Implementation

Account.java contains four private data fields, initially set to default values of 0. The no-arg constructor crates the default account with the current date attached to it. The next constructor sets the id and balance of the account.

Following are mutators and accessors for the id, balance, and annualInterestRate values for the account. getDateCreated sets the account creation date as the current date and time, as well as assigns it to a String value for display.

getMonthlyInterestRate and getMonthlyInterest methods perform the calculations required to display the proper percentages and values for the monthly interest.

withdraw and deposit are both methods designed to manipulate the balance data field based on user input.

The last method, main, is designed to test the Account class to ensure that the code operates correctly.