

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

**<Assignment 11-02>**

Document Version: 0.1

Version Date: June 26, 2022

Created By: David Duron

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Rationale |
| 0.1 | 2022 JUN 26 | David Duron | Submit Assignment |

# Document Purpose

The purpose of this document is to define the Employee Class.

# Technical Specifications

## Purpose of Technical Implementation

The purpose of the Employee subclass is to create an object that stores information relevant to the employee such as their office phone number, salary, date of hire; and includes the information from the Person super class.

## Technical Implementation Components

This class has four methods that display relevant information of the object, three methods to set properties of the object, and one constructor that requires arguments to populate all properties.

**Properties**

1. Office: this property stores the employee’s office phone number
2. Salary: this property stores the employee’s salary amount
3. dateHired: this property stores the employee’s hired date

**Methods**

1. getOffice(): returns the office phone number
2. getSalary(): returns the salary amount in a two decimal precision formatted as a String
3. getDateHired(): returns the hired date in a month/day/year format
4. setOffice(int office): setters for office property
5. setSalary(double salary): setters for salary property
6. setDateHired(): takes no arguments, uses current epoch timestamp
7. setDateHired(long epoch): takes no arguments, uses specified epoch timestamp
8. toString(): displays all information of the object in a readable format

**Constructors**

The Employee class is constructed with six arguments. Each is assigned to the properties of this class and those inherited from the parent class.

Employee employee\_object\_variable = new Employee(String name, String address, String phone, String email, int office, double salary)

## Technical Implementation Pseudocode

Ask user for input that will be used to construct the Employee object

Create instance of the Employee class with required arguments

Use the appropriate get method to display office, salary, date hired, or see all information in a readable format.

Use the setDateHired method with argument to specify specific hire date

Use the setSalary or setOffice to change the stored information of the object

End