Module 5 Lab 5B

Name: Clayton Black
Date: 10-22-2019
Assignment Name: Module 5 Lab 5B
Assignment Brief: recursion
Sources:

- https://ccse.kennesaw.edu/fye/pseudocode/pseudocodeguide.php

Main

```
CLASS Main
BEGIN

    METHOD Main
    BEGIN
        CREATE HourlyEmployee[2]

        CREATE firstName, lastName, id, address, city, state, hourlyRate, hoursWorked

        FOR i ← 0; i < 2; i++
            firstName ← getStringFromUser("Employee #" + i + "'s First Name: ")
            lastName ← getStringFromUser("Employee #" + i + "'s Last Name: ")
            id ← getStringFromUser("Employee #" + i + "'s id: ")
            address ← getStringFromUser("Employee #" + i + "'s address: ")
            city ← getStringFromUser("Employee #" + i + "'s city: ")
            state ← getStringFromUser("Employee #" + i + "'s state: ")

            hourlyRate = getDoubleFromUser("Employee #" + i + "'s Hourly Rate: ")
            hoursWorked = getDoubleFromUser("Employee #" + i + "'s Hours Worked: ")

            employee[i] = new HourlyEmployee(firstName, lastName, id, address, city, state, hourlyRate, hoursWorked)
        ENDFOR

        FOREACH employees as employee
            earnings = employee.earnings()
            print employee
            print "Employee Earnings: " + earnings
        ENDFOREACH
    END METHOD

    METHOD getStringFromUser(parameters: question)
    BEGIN
        PRINT question
        READ input
        return input
    END METHOD

    METHOD getDoubleFromUser(parameters: question)
    BEGIN
        WHILE TRUE
            PRINT question
            READ input
            IF input is a double THEN
                return input
            ELSE
                PRINT "Input Must be a double"
            ENDIF
        ENDWHILE

    END METHOD
END CLASS
```

Employee

```
ABSTRACT CLASS Employee
BEGIN
    CREATE firstName, lastName, employeeID, employeeStreetAddress, employeeCity, employeeState
    CONSTRUCTOR Employee(parameters: firstName, lastName, employeeID, employeeStreetAddress, employeeCity, employeeState)
    BEGIN
        this.firstName = firstName
        this.lastName = lastName
        this.employeeID = employeeID
        this.employeeStreetAddress = employeeStreetAddress
        this.employeeCity = employeeCity
        this.employeeState = employeeState

    END CONSTRUCTOR

    ABSTRACT METHOD earnings()
END ABSTRACT CLASS
```

HourlyEmployee

```
CLASS HourlyEmployee EXTENDS Employee
    CREATE hourlyRate, hoursWorked

    CONSTRUCTOR
    BEGIN
        super(firstName, lastName, employeeID, employeeStreetAddress, employeeCity, employeeState)
        this.hourlyRate = hourlyRate
        this.hoursWorked = hoursWorked
    END CONSTRUCTOR

    METHOD earnings()
    BEGIN
        RETURN hourlyRate*hoursWorked
    END METHOD

    METHOD toString()
    BEGIN
        return "HourlyEmployee{" +
                "hourlyRate=" + hourlyRate +
                ", hoursWorked=" + hoursWorked +
                ", firstName='" + firstName + '\'' +
                ", lastName='" + lastName + '\'' +
                ", employeeID='" + employeeID + '\'' +
                ", employeeStreetAddress='" + employeeStreetAddress + '\'' +
                ", employeeCity='" + employeeCity + '\'' +
                ", employeeState='" + employeeState + '\'' +
                '}'
    END METHOD
END CLASS
```