Object Oriented Programming
Lab Manual 03

1. A Bank

Look at the `Account` class `Account.java` and write a `main` method in a different class to briefly experiment with some instances of the `Account` class.

- Using the `Account` class as a base class, write two derived classes called `SavingsAccount` and `CurrentAccount`. A `SavingsAccount` object, in addition to the attributes of an `Account` object, should have an interest variable and a method which adds interest to the account.
  A `CurrentAccount` object, in addition to the attributes of an `Account` object, should have an overdraft limit variable. Ensure that you have overridden methods of the `Account` class as necessary in both derived classes.
- Now create a `Bank` class, an object of which contains an array of `Account` objects. Accounts in the array could be instances of the `Account` class, the `SavingsAccount` class, or the `CurrentAccount` class. Create some test accounts (some of each type).
- Write an update method in the bank class. It iterates through each account, updating it in the following ways: Savings accounts get interest added (via the method you already wrote); CurrentAccounts get a letter sent if they are in overdraft.
- The `Bank` class requires methods for opening and closing accounts, and for paying a dividend into each account.

Hints:

- Note that the balance of an account may only be modified through the `deposit(double)` and `withdraw(double)` methods.
- The `Account` class should not need to be modified at all.
- Be sure to test what you have done after each step.

2. Employees

Create a class called `Employee` whose objects are records for an employee. This class will be a derived class of the class `Person` which you will have to copy into a file of your own and compile. An employee record has an employee's name (inherited from the class `Person`), an annual salary represented as a single value of type `double`, a year the employee started work as a single value of type `int` and a national insurance number, which is a value of type `String`.

Your class should have a reasonable number of constructors and accessor methods, as well as an `equals` method. Write another class containing a `main` method to fully test your class definition.