**COSC 2050 (JAVA Programming I)**

Assignment 3, (Due 09/19/2017, before class)

**Work with customer and employee data**

## Console

Welcome to the Person Tester application

Create customer or employee? (c/e): c

Enter first name: Frank

Enter last name: Jones

Enter email address: frank44@hotmail.com

Customer number: M10293

You entered:

Name: Frank Jones

Email: frank44@hotmail.com

Customer number: M10293

Continue? (y/n): y

Create customer or employee? (c/e): e

Enter first name: Anne

Enter last name: Prince

Enter email address: anne@murach.com

Phone number: 111-11-1111

You entered:

Name: Anne Prince

Email: anne@murach.com

Phone number: 111-11-1111

Continue? (y/n): n

## Operation

* The application prompts the user to enter a customer or an employee.
* If the user selects customer, the application asks for name, email, and customer number.
* If the user selects employee, the application asks for name, email, and Phone number.
* When the user finishes entering data for a customer or employee, the application displays the data that the user entered.

## Specifications

* Create an abstract Person class that stores first name, last name, and email address. This class should provide a no-argument constructor, get and set methods for each piece of data, and it should override the toString method so it returns the first name, last name, and email fields in this format:

Name: Frank Jones  
Email: frank44@hotmail.com

In addition, it should contain an abstract method named getDisplayText that returns a string.

* Create a class named Customer that inherits the Person class. This class should store a customer number, it should provide get and set methods for the customer number, it should provide a no-argument constructor, and it should provide an implementation of the getDisplayText method. The getDisplayText method should return a string that consists of the string returned by the toString method of the Person class appended with the Customer number like this:

Name: Frank Jones  
Email: frank44@hotmail.com  
Customer number: M10293

* Create a class named Employee that inherits the Person class. This class should store a phone number, it should provide get and set methods for the phone number, it should provide a no-argument constructor, and it should provide an implementation of the getDisplayText method. The getDisplayText method should return a string that consists of the string returned by the toString method of the Person class appended with the Employees phone number like this:

Name: Anne Prince  
Email: anne@murach.com  
Phone number: 111-11-1111

* Create a class named PersonApp that prompts the user as shown in the console output. This class should create the necessary Customer and Employee objects from the data entered by the user, and it should use these objects to display the data to the user. To print the data for an object to the console, this application should use a static method named print that accepts a Person object.
* Use the Validator class or a variation of it to validate the user’s entries.