

# 1 Instructions

When you are finished submit all your work through the MyClasses page for this class. Create a directory called Homework05, put each programming exercise into its own subdirectory of this directory, zip the entire Homework05 directory up into the file Homework05.zip, and then submit this zip file to Homework #5.

Make sure that you:

- Follow the coding and documentation standards for the course as published in the MyClasses page for the class.
- Check the contents of the zip file before uploading it. Make sure all the files are included.
- Make sure that the file was submitted correctly to MyClasses.

All class structures are to have their own guarded specification file (.h) and implementation file (.cpp) that has the same name as the class. No inline coding in the .h files. In addition you must create a make file that compiles and links the project on a Linux computer with a Debian or Debian branch flavor.

# 2 Programming Exercise

This is a continuation of Lab #5, so make sure that you have completed all of the tasks in that lab before proceeding. You are going to add in another class structure in the inheritance chain. A retail store has a preferred customer plan where customers may earn discounts on all their purchases. The amount of a customer's discount is determined by the amount of the customer's cumulative purchases in the store.

- When a preferred customer spends \$500, he or she gets a 5% discount on all future purchases.
- When a preferred customer spends \$1,000, he or she gets a 6% discount on all future purchases.
- When a preferred customer spends \$1,500, he or she gets a 7% discount on all future purchases.
- When a preferred customer spends \$2,000 or more, he or she gets a 10% discount on all future purchases.

Design a class named `PreferredCustomer`, which is derived from the `CustomerData` class you created in Lab #5. The `PreferredCustomer` class should have the following member variables:

- purchasesAmount (a double)
- discountLevel (a double)

The purchasesAmount variable holds the total of a customer's purchases to date. The discountLevel variable should be set to the correct discount percentage, according to the store's preferred customer plan. This class should also have the functions PrintRecord and getRecord. These two functions must also be set up for dynamic binding. The PreferredCustomer class should call the PrintRecord and getRecord functions from the parent class to do its portion of the printing or string creation, you do not want to rewrite the code you already wrote for the base class.

**Input Validation:** *Do not accept negative values for any sales figures.*

You will now write a main program that will illustrate the dynamic binding of this set of class structures. You will allow the user to select which class they wish to insert into an array of base class pointers and allow them to print out the array of structures. So there is no way that the compiler will know which classes are placed where in the array, hence dynamic binding.

Specifically, you will create an array of 10 pointers to the PersonData (base class) type. You will also have an integer databaseSize that will hold the number of objects you inserted into the array. When the program starts the user will see the following menu.

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice:
```

If the user selects 1, 2, or 3 the program will ask the user for all the needed information, load the appropriate object with the data and then put this new object at the end of the array as well as increment the databaseSize value. Options 4 and 5 will print out the contents of the array (just the entries that have been loaded) to the console, either in condensed format or nice format. A run is below.

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 2

Input Last Name: Doe
Input First Name: Jack
Input Street Address: 123 River Dr.
Input City: Nowhere
Input State: PA
Input Zip Code: 12345
Input Phone Number: 111-222-3333
Input Customer Number: 77777
```

Do they wish to be on the mailing list? (Y/N): n

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 1
```

```
Input Last Name: Doe
Input First Name: Jane
Input Street Address: 333 Kiln St.
Input City: Somewhere
Input State: MD
Input Zip Code: 22222
Input Phone Number: 444-555-6666
```

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 4
```

```
Doe, Jack
123 River Dr.
Nowhere, PA 12345
111-222-3333
Customer Number: 77777
On the Mailing List: No
```

```
Doe, Jane
333 Kiln St.
Somewhere, MD 22222
444-555-6666
```

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 5
```

```
Jack Doe / 123 River Dr. / Nowhere, PA 12345 / 111-222-3333 / C#: 77777 / ML: No
Jane Doe / 333 Kiln St. / Somewhere, MD 22222 / 444-555-6666
```

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 3
```

```
Input Last Name: Public
Input First Name: John
Input Street Address: 25 Gilmore St.
Input City: Elsewhere
Input State: MN
Input Zip Code: 55555
Input Phone Number: 777-888-9999
Input Customer Number: 34645
Do they wish to be on the mailing list? (Y/N): y
Input Purchases Amount: 572
```

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 4
```

```
Doe, Jack
123 River Dr.
Nowhere, PA 12345
111-222-3333
Customer Number: 77777
On the Mailing List: No
```

```
Doe, Jane
333 Kiln St.
Somewhere, MD 22222
444-555-6666
```

```
Public, John
25 Gilmore St.
Elsewhere, MN 55555
777-888-9999
Customer Number: 34645
On the Mailing List: Yes
Purchases Amount: 572
Discount Level: 5%
```

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 5
```

```
Jack Doe / 123 River Dr. / Nowhere, PA 12345 / 111-222-3333 / C#: 77777 / ML: No
Jane Doe / 333 Kiln St. / Somewhere, MD 22222 / 444-555-6666
John Public / 25 Gilmore St. / Elsewhere, MN 55555 / 777-888-9999 / C#: 34645 / ML: Yes /
Account: $572.000000 / Discount: 5.000000%
```

```
1. Add Personal Data Object
2. Add Customer Data Object
3. Add Preferred Customer Data Object
4. View Database Long Form
5. View Database Short Form
6. Quit
Choice: 6
```