

Lab Workbook 3

Software Development HDSWD

Instantiable Classes

12/05/12

Jonathan Meaney

For each problem make sure to use the same steps we have been using so far!

1. Problem Definition

- a. Define the problem in your own words
- b. Read the problem and determine what it is asking you to do. Highlight the keywords
- c. Describe the features of the application and what its goal is

2. Design - Overall Plan

- a. What objects and classes are you going to use
- b. What types of data will I be working with
- c. What is the input
- d. What processing will be done
- e. What will the output be
- f. What will be the components of the class be
 - i. What data members do you need
 - ii. What will your constructor do (default initialization, initialize to different values using overloaded constructors)
 - iii. What methods will you need (getters and setters for data members, what behaviour is needed, what will it do)

3. Implementation

- a. Create the classes
- b. Declare and create objects
- c. Provide input and display output

4. Testing

- a. Was the output correct?
- b. Did the program compile and run successfully
- c. Test the application with different values

Problem 1:

Define a new class named Temperature. The class has two methods toFahrenheit() and toCelsius() that return the temperature in the specified unit. Create the appropriate classes, data members, constructor or constructors and methods to implement this application.

Problem 2:

A student has a name, an id number and an email address. Define a class called Student to represent students. Implement two constructors, a default one and one that assigns values to each of the data members. Declare and create a Student object using each constructor, one with no arguments and one with three. Set the data of the no argument Student object using the setter methods. Display the details of each student.

Problem 3:

Extend the Circle class to have a calcCircumference() method and a displayDetails method which prints the values of the data members.

Problem 4:

Create a class called Car that represents cars. A car has the following attributes, year manufactured, miles driven, manufacturer name and model name. Create the appropriate data members, constructors and methods to implement the class. Also create a display method to display the details of the object. Since the distance driven is represented in miles create a method to display this data in kilometres.

Problem 5:

Create an application that simulates a bank account. Define a new class named Account which represents a persons bank account. An account has an account number, owner, balance and bank branch. Its possible to deposit and withdraw money from an account and also display the current balance and account details. Create two account objects lecturersAccount and studentAccount and initialize the data members to have default values. studentAccount should have a balance of 10,000 and lecturersAccount should have a balance of 0 with a bank branch located in the Cayman Islands. Withdraw 10,000 from studentAccount and deposit 10,000 into lecturersAccount. Display the balance and details of both accounts.