Lab 5

(Due date: Friday, 2/13/2009 in the Lab hours)

Your name:	Score:

1.Practice with Fig3.10 and Fig3.11 from the textbook (the following is the sample code for this project). Write down the important statements in the blank space.

```
// Fig. 3.10: GradeBook.java
// GradeBook class with a constructor to initialize the course name.

public class GradeBook
{
   private String courseName; // course name for this GradeBook

   // constructor initializes courseName with String supplied as argument
   public GradeBook( String name )
   {
      courseName = name; // initializes courseName
   } // end constructor

   // method to set the course name
   public void setCourseName( String name )
   {
      courseName = name; // store the course name
   } // end method setCourseName

   // method to retrieve the course name
   public String getCourseName()
   {
      return courseName;
   } // end method getCourseName

   // display a welcome message to the GradeBook user
   public void displayMessage()
   {
      // this statement calls getCourseName to get the
      // name of the course this GradeBook represents
      System.out.printf( "Welcome to the grade book for\n%s!\n",
         getCourseName() );
   } // end method displayMessage

} // end class GradeBook

// Fig. 3.11: GradeBookTest.java
// GradeBook constructor used to specify the course name at the
// time each GradeBook object is created.

public class GradeBookTest
{
   // main method begins program execution
   public static void main( String args[] )
   {
      // create GradeBook object
      GradeBook gradeBook1 = new GradeBook(
```

```
            "CS101 Introduction to Java Programming" );
         GradeBook gradeBook2 = new GradeBook(
            "CS102 Data Structures in Java" );

         // display initial value of courseName for each GradeBook
         System.out.printf( "gradeBook1 course name is: %s\n",
            gradeBook1.getCourseName() );
         System.out.printf( "gradeBook2 course name is: %s\n",
            gradeBook2.getCourseName() );
      } // end main

   } // end class GradeBookTest
```

2.Practice with Fig3.13 and Fig3.14 from the textbook (the following is the sample code for this project). Write down the important statements in the blank space.

```
// Fig. 3.13: Account.java
// Account class with a constructor to
// initialize instance variable balance.

public class Account
{
   private double balance; // instance variable that stores the balance

   // constructor
   public Account( double initialBalance )
   {
      // validate that initialBalance is greater than 0.0;
      // if it is not, balance is initialized to the default value 0.0
      if ( initialBalance > 0.0 )
         balance = initialBalance;
   } // end Account constructor

   // credit (add) an amount to the account
   public void credit( double amount )
   {
      balance = balance + amount; // add amount to balance
   } // end method credit

   // return the account balance
   public double getBalance()
   {
      return balance; // gives the value of balance to the calling method
   } // end method getBalance

} // end class Account

// Fig. 3.14: AccountTest.java
// Inputting and outputting floating-point numbers with Account objects.
import java.util.Scanner;

public class AccountTest
{
   // main method begins execution of Java application
   public static void main( String args[] )
   {
      Account account1 = new Account( 50.00 ); // create Account object
      Account account2 = new Account( -7.53 ); // create Account object
```

```
      // display initial balance of each object
      System.out.printf( "account1 balance: $%.2f\n",
         account1.getBalance() );
      System.out.printf( "account2 balance: $%.2f\n\n",
         account2.getBalance() );

      // create Scanner to obtain input from command window
      Scanner input = new Scanner( System.in );
      double depositAmount; // deposit amount read from user

      System.out.print( "Enter deposit amount for account1: " ); // prompt
      depositAmount = input.nextDouble(); // obtain user input
      System.out.printf( "\nadding %.2f to account1 balance\n\n",
         depositAmount );
      account1.credit( depositAmount ); // add to account1 balance

      // display balances
      System.out.printf( "account1 balance: $%.2f\n",
         account1.getBalance() );
      System.out.printf( "account2 balance: $%.2f\n\n",
         account2.getBalance() );

      System.out.print( "Enter deposit amount for account2: " ); // prompt
      depositAmount = input.nextDouble(); // obtain user input
      System.out.printf( "\nadding %.2f to account2 balance\n\n",
         depositAmount );
      account2.credit( depositAmount ); // add to account2 balance

      // display balances
      System.out.printf( "account1 balance: $%.2f\n",
         account1.getBalance() );
      System.out.printf( "account2 balance: $%.2f\n",
         account2.getBalance() );
   } // end main

} // end class AccountTest
```

3.Practice with Fig3.18 from the textbook (the following is the sample code for this project). Write down the important statements in the blank space.

```
// Fig. 3.18: NameDialog.java
// Basic input with a dialog box.
import javax.swing.JOptionPane;

public class NameDialog
{
   public static void main( String args[] )
   {
      // prompt user to enter name
      String name =
         JOptionPane.showInputDialog( "What is your name?" );

      // create the message
      String message =
         String.format( "Welcome, %s, to Java Programming!", name );

      // display the message to welcome the user by name
      JOptionPane.showMessageDialog( null, message );
```

```
    } // end main
} // end class NameDialog
```

4.Complete Exercise 3.11 on page 125.

5.Complete Exercise 3.12 on page 125.