Lab #2:

Problem.

Part A. Create a Cake class:

1. Your cake class should have a private String instance variable named flavor.
2. A constructor with no arguments which sets flavor to Chocolate.
3. A constructor which accepts has a String parameter and sets flavor to that string.
4. A method display, which prints the flavor.

Part B. Create a class called ShortCake (FYI, Shortcakes are often vanilla with a berry topping).

ShortCake should inherit from cake and have the following specifications:
1. A private String instance variable named toppings.
2. A constructor that takes two parameters, a String value for the cake flavor, and a String value for toppings. This constructor MUST explicitly specify the correct super constructor. It should also set the topping instance variable.
3. A public get method to return toppings.
4. An overridden method, display, that prints out the cake flavor (Note: There is no getFlavor method in Cake and flavor is private – you need to call the superclass’s display method, but think about how to call your super class’s display method if we already have an overridden method with that name in our own class!). Print the topping as well. Remember to use @Override.

Part C. Create a class called TestShortCake which has two methods, main() and cakeInfo().

Within the main method:
1. Create a ShortCake Object passing in Vanilla and Strawberry as the flavor and topping.
2. Call a static method cakeInfo that you will create in this class. Pass in the ShortCake object you just created.

The cakeInfo() method has the following:
1. It’s a static method which has a Cake parameter (Note: It must be Cake not ShortCake). You can decide what the return type should be.
2. Call the method display() to print the flavor and topping (This will work because the Cake object has the method display and it will use the overridden one).
3. If the cake object is a ShortCake (Hint: use instanceof to check), print only the topping. Note: You will need to think about how to print the topping. How do we have access to a private variable? How do we have access to a method in ShortCake if our object is of type Cake!

See my output below:

Output:
The cake flavor is: Vanilla
The topping is: Strawberry
Topping: Strawberry

Submission: Zip up a folder containing the 3 java files and one output file. Submit to the dropbox labeled Lab2.