The University of Winnipeg

Introduction to Java Programming

Andre Specht

**Assignment 2: setters/getters, constructors and files Weight: 20%**

**Project 1: tossing coins**

Write a program that toss a coin. The main class (Coin) should contain the following:

* A String named *sideUp*. The sideUp field will hold either 'heads' or 'tails' indicating the side of the coin that is facing up.
* A no-arg constructor that randomly determines the side of the coin that is facing up ('heads' or 'tails') and initializes the sideUp field accordingly.
* A void method named toss() that simulates the tossing of the coin. When the toss method is called, it randomly determines the side of the coin that is facing up ('heads' or 'tails') and sets the sideUp field accordingly.
* A method named *getSideUp* that returns the value of the sideUp field.

**System requirements**

The program should create an instance of the class and display the side that is initially facing up. Then, use a loop to toss the coin 10 times. Each time the coin is tossed, display the side that is facing up. The program should keep count of the number of times heads is facing up and the number of times tails is facing up, and display those values after the loop finishes.

**Program output (example)**

Coin is current on tails

Tossing coin...

#1 Heads

#2 Tails

#3 Heads

#4 Tails

#5 Heads

#6 Heads

#7 Heads

#8 Heads

#9 Heads

#10 Heads

Heads wins: 8

Tails wins: 2

**Rubrics**

* Code in a single Java file.
* All required class members are implemented.
* Both constructor and toss() method generate random results (heads/tails).
* Formatted output.
* The program works and delivers the expected results.
* Code style (naming conventions, indentation, meaningful comments, etc.).

**Project 2: mailing list**

Follow the steps below to develop a program to store customer data.

Design a class named Person with fields for holding a person’s name, address, and telephone number. Write one or more constructors and the appropriate mutator and accessor methods (setters and getters) for the class’s fields.

Next, design a class named Customer, which extends the Person class. The Customer class should have a field for a customer number and a Boolean field indicating whether the customer wishes to be on a mailing list. Write one or more constructors and the appropriate mutator and accessor methods (setters and getters) for the class’s fields. Demonstrate an object of the Customer class in a simple program (a third class with the main() method).

Implement a toString() method in Customer class to print the data about customers (the object).

You should be able to store data using both constructor and mutator methods (setters).

**Program output (example)**

Number: 1859

Name: John Smith

Address: 123 Notre Dame Ave

Phone: (204) 234-5678

Mailing list: Yes

**Rubrics**

* Code in a single Java file.
* All required classes (3) and class members (constructors, setters and getters) are implemented.
* Encapsulation is present in both classes, Person and Customer.
* toString() method is implemented in Customer class.
* Program works and delivers the expected results.
* Code style (naming conventions, indentation, meaningful comments, etc.).

**Project 3: anagrams**

Write a program that lists all anagrams found in a *txt* file (*words.txt*).

**System requirements**

1. Ask the user to enter a valid English word.
2. Convert the word into a key.
3. Read all words listed in the *txt* file.
4. Convert each word into a key that will be compared to the one entered by the user.
5. Store anagrams found in an ArrayList.
6. Print the results (the anagrams found, excluding the user’s input).

**Rubrics**

* Code in a single Java file.
* Method to generate a keyword.
* Program opens and reads the txt file.
* Program works and delivers the expected results.
* Word entered by the user is not included in the output.
* Code style (naming conventions, indentation, meaningful comments, etc.).

**Submission guidelines**

Upload your files to the Dropbox folder (Nexus 🡪 Assignments 🡪 Assignment 3) as a zipped folder (with 3 Java files). This file should be named as follows: *your\_full\_name\_java\_assignment\_3*.