CSE 102L Computer Programming Laboratory – Exercise 6

Due Date: 23:59 Sunday April 30th

Disclosure

You will submit your file to an assignment that is given through MS teams. Your filename should be *Ex6_yourStudentNumber.java*. Submissions made after the deadline will not be accepted, be sure to submit your work before the due date and make sure to click turn in button. Your code will be automatically controlled, so be sure to have the same class, method, variable names as described here. Failure to do so may result in you receiving 0 from this exercise. **All classes should be written to a single Java file. In a single java file there can only be single public class. Do NOT use Inner Classes. Be careful naming your file. If your editor inserts the file into a package, remove that line from the file but do NOT delete the import statements.**

Write set of classes according to the following specifications. Declare all attributes as **private** if not requested otherwise and use camelCase formatting for attributes.

1. Product – Abstract, implements Comparable

- Attributes:
  - name: String
  - price: double
- Methods:
  - Constructor that takes *name* and *price*
  - Accessors for attributes
  - Compares according to *price*
  - toString(): String – returns String in the format:
    - ClassName " [name= " + name + ", price= " + price + "]"

2. Book – Abstract, child of Product

- Attributes:
  - author: String
  - pageCount: int
- Methods:
  - Constructor that takes *name*, *price*, *author*, and *pageCount*
  - Accessors for attributes

3. ReadingBook – child of Book

- Attributes:
    - genre: String
- Methods:
    - Constructor that takes *name, price, author, pageCount,* and *genre*
    - Accessor for attribute

4. ColoringBook – child of Book, implements Colorable

- Attributes:
    - color: String
- Methods:
    - Constructor that takes *name, price, author, pageCount*
    - Accessor for attribute

5. ToyHorse – child of Product implements Rideable

6. Bicycle – child of Product implements Colorable and Rideable

- Attributes:
    - color: String
- Methods:
    - Accessor for attribute

7. User

- Attributes:
    - username: String
    - email: String
    - payment: PaymentMethod
    - cart: collection of product
- Methods:
    - Constructor that takes *username* and *email*
    - Accessor for *username* and *email*
    - *Mutator* for payment
    - getProduct(index: int): Product
    - removeProduct(index: int): None
    - addProduct(product: Product): None
    - setPayment(payment: PaymentMethod): None
    - purchase(): None – sums prices for products and uses *payment*'s pay method. If successful, clears cart

8. CreditCard – implements PaymentMethod

- Attributes:
    - cardNumber: long
    - cardHolderName: String
    - expirationDate: **java.util.**Date (not the one from sql, be careful)
    - cvv: int
- Methods:
    - Constructor that takes *cardNumber, cardHolderName, expirationDate,* and *cvv*

9. PayPal – implements PaymentMethod

- Attributes:
    - username: String
    - password: password
- Methods:
    - Constructor that takes *username* and *password*

+50 points if you implement a known encrpytion algorithm (You must at least get 60 points from rest of the exercise to recieve this additional point)

1. Colorable – Interface:

- Methods:
    - paint(color: String): None

2. Rideable – Interface:

- Methods:
    - ride(): None

3. PaymentMethod – Interface:

- Methods:
    - pay(amount: double): boolean