East West University
Department of Computer Science and Engineering

CSE 110: LAB 07 (Lab Exam)
Course Instructor: Dr. Mohammad Rezwanul Huq

Student ID	
Student Name	

Question 1 (4 marks):

A perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself. For example, 6 is a Perfect number. Its proper divisors are 1, 2 and 3, and 1 + 2 + 3 = 6.

Write a program in Java that takes a series of integers and prints whether the number is a Perfect number. Use scanner class to take the inputs and input is terminated when the user enters 0. A sample input and output for this program is given below.

```
Sample Input:          Sample Output:
6                      6 is a Perfect number
30                     30 is not a Perfect number
28                     28 is a Perfect number
0
```

Question 2 (6 marks):

The following diagram shows relationships among three classes: Customer, Transaction and Product.

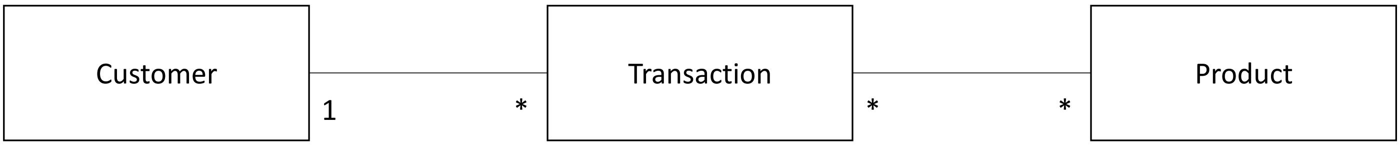

Figure 1: Relationship Diagram

Define these classes as per the following definitions.

Customer	**Transaction**	**Product**
- customerId: int - customerName: String - customerAddr: String	- transactionId: int - transactionDate: Calendar - productList: Product[] - productQuantity: int[] - customer: Customer - numberOfProducts: int	- productId: int - productName: String - productPrice: double

// add constructors
// add appropriate getters and setters
+ toString(): String
	// add constructors
	// add appropriate getters and setters
	+ toString(): String
	+ addProduct(Product): void
	+ dropProduct(authorId): void
	+ addCustomer(Publisher): void
	+ getTotalAmount(): double
		// add constructors
		// add appropriate getters and setters
		+ toString(): String

You must write a Main class that should have the following public static variables and methods:

i) Keep a public static type array of Customers to hold all customers. Choose reasonable size for the array.

(Example: `public static Customer[] customers = new Customer[10];`)

ii) Do the same for holding Transactions and Products list.

iii) Define a static method `searchTransactions(transactionId)` that takes the transaction id as the parameter and prints all details about that transaction including customer name and all product list and their quantities.

iv) Define a static method `updateProduct(productId)` that takes the product id as the parameter and prompts the user to provide modified value for product name and price.

v) (Bonus Task): Define a static method `showProductsByCustomer(customerId)` that takes the customer id as the parameter and prints the product name and product price of all products which are bought by that customer considering all transactions.

Inside your main method, you must do the followings:

i) Create a few objects of Customer, Transactions and Product type and store them appropriately.

ii) Add a few products and their quantities in the product list and product quantity array of several transactions by calling `addProduct(Product)` instance method of Transaction class.

iii) Drop a product from the product list of a transaction by calling `dropProduct(productId)` instance method of Transaction class. The corresponding entry in productQuantity array should be also dropped.

iv) Add a customer for several transactions by calling `addCustomer(Customer)` instance method of the Transaction class.

v) You must show the total amount in a transaction by using `getTotalAmount()` instance method of Transaction class.

`vi)` Call the above-mentioned methods - `searchTransactions(transactionId)`, `updateProduct(productId)` and `showProductsByCustomer(customerId)` to demonstrate their functionalities.