|  |  |  |
| --- | --- | --- |
| **LAB211 Assignment** |  |  |
| **Lab 4** |  |
| **LOC:** | **60** |
|  |  |

**Title**

Account

**Background**

N/A

**Program Specifications**

A cash processing company has a class called Account used to process transactions:

|  |  |
| --- | --- |
| **Method/Constructor** | **Description** |
| public Account(Client c) | constructs an account using client information |
| public boolean process(Transaction t) | processes the next transaction, returning true if transaction was approved, false otherwise |

Account objects interact with Transaction objects, which have many methods including:

|  |  |
| --- | --- |
| **Method/Constructor** | **Description** |
| public int value() | returns the value of this transaction in pennies (could be negative, positive or zero) |

The company wishes to create a slight modification to the Account class that filters out zero-valued transactions. Design a new class called FilteredAccount whose instances can be used in place of an Account object but which include the extra behavior of not processing transactions with a value of 0. More specifically, the new class should indicate that a zero-valued transaction was approved but shouldn't call the process method in the Account class to process it. Your class should have a single constructor that accepts a parameter of type Client, and it should include the following method:

|  |  |
| --- | --- |
| **Method/Constructor** | **Description** |
| public double percentFiltered() | returns the percent of transactions filtered out (between 0.0 and 100.0); returns 0.0 if no transactions submitted |

Assume that all transactions enter the system by a call on the process method described above.

Write a class with main method to Demo all above classes.