**[เฉลย] Lab6-Static Fields**

กำหนดให้ใบสั่งซื้อ (Order) มีข้อมูลที่ต้องเก็บ คือ เลขใบคำสั่งซื้อ วันที่สั่งซื้อ จำนวนเงินที่ซื้อ การสร้างวัตถุใหม่ทุกครั้งให้เลขใบคำสั่งซื้อมีการรันเลขต่อไป เช่น

Order o1 = new Order(LocalDate.of(2018, Month.FEBRUARY, 14),2500);

// วัตถุ ของ Order ที่สร้างขึ้น จะมีรหัสเท่ากับ “Order-0001”

System.out.print( o1.getOrderInformation());

// ID: Order-0001 Date: 2018-02-14 Amount: 2,500.00

Order e2 = new Order( LocalDate.of(2018, Month.FEBRUARY, 15),3500);

// วัตถุ ของ ของ Order ที่สร้างขึ้น จะมีรหัสเท่ากับ “Order-0002” Date: 2018-02-15 Amount: 3,500.00

System.out.print( o2.getOrderInformation());

//------------------------------------------------------------------------------------------------

import java.time.LocalDate;

import java.time.format.DateTimeFormatter;

public class Order {

private static int orderCounter = 0;

private String id;

private LocalDate date;

private double amount;

public Order() {

orderCounter++;

}

public Order(LocalDate date, double amount) {

orderCounter++;

this.date = date;

this.amount = amount;

this.id = String.format("Order-%04d", orderCounter);

}

public String getOrderInformation() {

DateTimeFormatter dateformat = DateTimeFormatter.ofPattern("dd-MM-yyyy");

String dateForm = date.format(dateformat);

String info = String.format("ID: %s Date: %s Amount: %.2f", this.id, dateForm, this.amount);

return info;

}

public String getId() {

return this.id;

}

public void setId() {

this.id = String.format("Order-%04d", orderCounter);

}

public LocalDate getDate() {

return this.date;

}

public void setDate(LocalDate date) {

this.date = date;

}

public double getAmount() {

return this.amount;

}

public void setAmount(double amount) {

this.amount = amount;

}

}

//------------------------------------------------------------------------------------------------

import java.time.LocalDate;

import java.time.Month;

import java.time.format.DateTimeFormatter;

public class Main {

public static void main(String[] args) {

Order o1 = new Order(LocalDate.now(), 2500);

System.out.println(o1.getOrderInformation());

Order e2 = new Order( LocalDate.of(2018, Month.FEBRUARY, 15),3500);

System.out.println( e2.getOrderInformation());

}

}