

แบบฝึกปฏิบัติการครั้งที่ 13 : File and exception handling

ให้นิสิตดู Java API ในส่วนของคลาส String และ ArrayList เพื่อหา method ที่ช่วยให้เราทำงานได้ง่ายขึ้น

1. จงเขียนโปรแกรมเพื่ออ่านข้อความที่ผู้ใช้พิมพ์แล้วนำไปเขียนลงไฟล์แบบ sequential text file หยุดเมื่อผู้ใช้ป้อนคำว่า quit จากนั้นให้เปิดอ่านข้อมูลจากไฟล์เพื่อนับตัวอักขระ คำ และบรรทัดในไฟล์นั้นแสดงผลทางจอภาพ

หมายเหตุ การทดสอบโปรแกรม ขอให้พิมพ์ข้อความที่เว้นวรรคด้วยการเคาะ 1 space เท่านั้น และการนับตัวอักขระ จะนับเฉพาะ space ไม่นับรวม '\n'

ไม่ต้องเขียนจัดการ exception เอง แค่ประกาศไว้ก็พอ

ตัวอย่างการรันครั้งที่ 1	ตัวอย่างการรันครั้งที่ 2
<pre>Hi John How are you?? quit Total characters : 20 Total words : 5 Total lines : 2</pre>	<pre>Hi John I Love You quit Total characters : 15 Total words : 5 Total lines : 4</pre>

2. จงเขียนโปรแกรมเพื่อรับข้อความ 1 ชุด และตรวจสอบว่า มีคำที่ไม่อยู่ในชุดคำศัพท์ (ซึ่งคัดลอกมาจาก <http://www.cs.duke.edu/~ola/ap/linuxwords> และสร้างเป็นไฟล์ wordlist.txt ให้แล้ว) หรือไม่ และแสดงคำที่ไม่มีในไฟล์ wordlist.txt เป็นผลลัพธ์ หากทุกคำในข้อความมีอยู่ในไฟล์ wordlist.txt ให้แสดงคำว่า N/A

ตัวอย่างการรันครั้งที่ 1	ตัวอย่างการรันครั้งที่ 2
<pre>Enter a sentence: this is fantastic Words not contained: N/A</pre>	<pre>Enter a sentence: this is awesome prog Words not contained: prog</pre>

ให้เขียน try-catch จัดการ exception เอง

hint: อ่านจากไฟล์มาเก็บไว้ใน array list ก่อน แล้วค่อยเอาไปใช้ทำงานต่อ

## 3. ธนาคารแห่งหนึ่ง มีไฟล์แบบ sequential text file ดังนี้

1. Master file เก็บข้อมูล account records บันทึกไว้ในไฟล์ master.txt ซึ่งประกอบด้วย account number, name และ balance
2. Transaction file เก็บข้อมูลการฝาก deposite (ตัวเลขเป็น +) และการถอน withdraw (ตัวเลขเป็น -) บันทึกไว้ในไฟล์ trans.txt ซึ่งประกอบด้วย account number และ amount โดยแต่ละบัญชีสามารถทำรายการ transaction ได้มากกว่า 1 ครั้ง

ทั้งสองไฟล์เชื่อมกันด้วย account number และเก็บข้อมูลแบบเรียงลำดับตาม account number จากน้อยไปมาก สมมติว่า ใน trans.txt จะมีแต่เลขบัญชีที่มีอยู่ใน master.txt เท่านั้น

master.txt	trans.txt
100 Alan Jones 348.50	100 27.50
300 Mary Smith 27.50	300 65.00
500 Sam Sharp 100.00	700 100.00
700 Suzy Green -14.50	700 100.00
	700 10.00

จึงสร้าง class TransactionRecord โดย object ของคลาสนี้ เก็บ account number และ amount of transaction ให้เขียน method set() get() สำหรับ 2 ตัวแปรนี้ตามความเหมาะสม

จงแก้ไข class AccountRecord ที่กำหนดให้ โดยเพิ่ม method combine(TransactionRecord t) เพื่อให้รวมเงินของ transaction เข้ากับเงิน balance ของ account number นั้น (ตัวเลข balance ที่คำนวณได้อาจเป็นเลขติดลบ ให้ถือว่าบัญชีนั้นเป็นหนี้ธนาคารอยู่)

```
public class AccountRecord {
    private int acctNo;
    private String name;
    private double balance;
    private int transCnt = 0; // นับว่าบัญชีนี้ทำรายการ transaction ไปกี่ครั้ง
    public AccountRecord (int acctNo, String name, double balance) {
        this.acctNo = acctNo;
        this.name = name;
        this.balance = balance;
    }
    public int getAcctNo() { return acctNo; }
    public String getName() { return name; }
    public double getBalance() { return balance; }
    public int getTransCnt() { return transCnt; }
}
```

จงสร้าง class FileMatch กำหนดให้สร้าง array list สำหรับเก็บ object ของ account record และอีกหนึ่ง array list สำหรับเก็บ object ของ transaction record โดยอ่านข้อมูลจาก master.txt เพื่อสร้าง object ของ account record เก็บใน array list และอ่านข้อมูลจาก trans.txt เพื่อสร้าง object ของ transaction เก็บใน array list แล้วทำการ matching เพื่อคำนวณและปรับปรุงค่า balance ของแต่ละบัญชี เสร็จแล้วนำข้อมูลบัญชีทั้งหมดมาเขียนลงไฟล์ newMaster.dat ซึ่งเป็นไฟล์แบบ Random access เพื่อความสะดวกในการเข้าถึงข้อมูลแต่ละบัญชี โดยแต่ละ account record ให้เก็บ account number (int), name (30 characters ถ้าข้อมูลชื่อยาวไม่ถึง 30 ตัวอักษรให้เติม space จนครบ 30), balance (double) และ transaction count (int) จากนั้นเข้าถึงไฟล์ newMaster.dat เพื่อประมวลผลข้อมูลและแสดงผลลัพธ์เป็นจำนวนบัญชีที่มีอยู่ ยอดเงินรวมของทุกบัญชี และจำนวนบัญชีที่ไม่มีการทำรายการ transaction เลย

ให้ใช้คำสั่ง try with resource และมี catch จัดการ exception เอง

ผลการรันโปรแกรมเป็นดังนี้

Total Account Record : 4 Total balance : 764.0 No transaction : 1 account.
--