

Assignment 8 HashMap

- ✚ ทดลองใช้คลาส HashMap ที่มีอยู่ใน JAVA ซึ่งจัดการโครงสร้างแบบ Hashing
 - Test Case ใช้กรณีเดียวกับ Assignment 7
- ✚ คลาส HashMap `//import java.util.HashMap`
 - ใช้ดำเนินงานเกี่ยวกับการจัดเก็บ และเรียกค้นข้อมูล โดยข้อมูลจะมีลักษณะเป็นคู่ คือ (key, value) มี key เป็นตัวค้น และ value เป็นค่าข้อมูล ที่จัดเก็บ
 - ใช้ key แปลงเป็นตำแหน่งข้อมูล(index)ที่จะเก็บ **ไม่มีการเรียงลำดับข้อมูล**
 - ไม่สามารถใส่ object ที่มีคีย์ที่ซ้ำได้ (ถ้าซ้ำจะทับตัวเดิม)
 - สามารถใช้ For Each ในการเข้าถึงข้อมูลทีละตัวได้
 - `for (String itr : dict.keySet()) { ต้องใช้ dict.get(itr) เพื่อตีความหมาย }`
 - ไม่สามารถดึงข้อมูลออกมาในลักษณะเรียงลำดับได้
- ✚ ตัวอย่างคำสั่งที่เกี่ยวข้องกับคลาส HashMap
 - การจองตัวแปรในคลาส HashMap (ตัวหน้าเป็นคีย์ , ตัวหลังเป็นข้อมูล)
 - การตรวจสอบสมาชิกว่ามีอยู่หรือไม่ `dict.containsKey(key)`
 - การเพิ่มข้อมูล `dict.put(key, x);`
 - ดึงข้อมูลตัวที่ตรงกับคีย์ `x = dict.get(key)`
 - การลบข้อมูล `dict.remove(key);`
 - นับจำนวนข้อมูล `size() i = dict.size();`

คำแนะนำ Assignment 8

- สร้าง/ดัดแปลง Assignment 7 ใหม่เปลี่ยนเป็นใช้คลาส HashMap
 - ออกแบบโครงสร้างข้อมูลให้มีฟิลด์ word เป็นและ mean (type ให้รวมเข้าไปใน mean)
- สร้างคลาส dictMap สำหรับรับ โดยใช้ word เป็น key และ ArrayList ของ mean เป็น value
 - กำหนดตัวแปร dict สำหรับเก็บข้อมูลทั้งหมดเป็น ชนิด HashMap

HashMap <String, ArrayList<String>> dict = new HashMap <String, ArrayList<String>>();

- สร้างเมธอดสำหรับอ่านข้อมูลจากไฟล์ทีละ 1 บรรทัด
 - ส่งบรรทัดที่อ่านได้ไปสร้างmap

```
buff = buff.trim().replaceAll("\\s+", " ");
String[] str = buff.split(",");
String word = str[0].toLowerCase();
String mean = str[1] + "(" + str[2] + " ";
if (dict.containsKey(word)) {
    ArrayList <String> z = dict.get(word);
    if (!z.contains(mean)) {
        z.add(mean);
        dict.put(word, z); }
} else {
    ArrayList <String> z = new ArrayList <String>();
    z.add(mean);
    dict.put(word, z); }
}
```

/ กรณีที่นำ Bnode มาใช้เป็น Value */
// Bnode x = new Bnode(buff);
// if (dict.containsKey(x.word))
// Bnode z = dict.get(x.word);
// if (!z.mean.contains(x.mean.get(0)))
// z.mean.addAll(x.mean);
// dict.put(x.word,z);

// dict.put(x.word, x);*

- สามารถนำ Bnode ที่ใช้ใน Assign7 มาใช้เป็น value ได้เลยโดยไม่ต้องสร้างอาร์เรย์ของคำแปล
HashMap <String, Bnode> dict = new HashMap <String, Bnode> ();



คำแนะนำ Assignment 8

- ตัวอย่างการค้นหาและแสดงผล

```
if (dict.containsKey(keystr)) { //ค้นหา key ใน dict
    ArrayList <String> z = dict.get(keystr); //Bnode z = dict.get(keystr);
    System.out.printf("%s have %d meaning\n", keystr, z.size()); //ใช้ z.mean แทน z
    for (int i = 0; i < z.size(); i++) //z.mean.size()
        System.out.println(i + 1 + ") " + z.get(i)); // z.mean.get(i)
} else
    System.out.printf("%s not found\n", keystr);
```

- ตัวอย่างการวนรอบทุกตัวใน dict

```
for (String itr : dict.keySet()) {// แต่ละรอบจะใช้ dict.get(itr) เป็นอาร์เรย์ของคำแปล
```