**[โจทย์] Lab7-Array of Objects & Aggregation**

ข้อ 1. ให้สร้าง คลาส Player มีฟิลด์ id, name, และ score แทน รหัส ชื่อ คะแนน ตามลำดับ

แล้วให้สร้างคลาส Team มีอะเรย์ของผู้เล่น ไม่เกิน 15 คน ถ้าเราเพิ่มเกินจะฟ้องข้อผิดพลาดว่าเต็มแล้ว

เพื่อให้สามารถประมวลผลคำสั่งต่อไปนี้

Team m = new Team ();

Player s1= new Player (“6011”,”John”,8900);

Player s2= new Player (“6012”,”Bo”,5000);

m.addNew Player (s1); //

m.addNew Player (s2); //

int count = m.getNumberOf Player (); // คืนจำนวนนักเรียน คือ 2 คน

Player [] players = m.findAllPlayers (); // คืนอะเรย์ของนักเรียนทุกคน

Player[ ] ls = new Player [2];

ls[0] = new Player (“602”,”Koi”,60);

ls[1] = new Player (“603”,”Mono”,10);

m.merge(ls);

double avg = m.meanOfScore(); จะคืนค่าเฉลี่ยคะแนนของผู้เล่น

Player p = m.findByName(“Mono”);

จะคืนค่าของตัวแปรอ้างอิง เพื่อให้ p.getName() มีค่าเป็น “Mono” และ p.getScore() มีค่าเป็น 10

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

ข้อ 2. ให้สร้าง คลาส Student มีฟิลด์ id, name, และ score แทน รหัส ชื่อ คะแนน ตามลำดับ

แล้วให้สร้างคลาส MathSection มีอะเรย์ของนักเรียน Student กำหนดให้มีขนาดไม่เกิน 20 คน ถ้าเราเพิ่มเกินจะฟ้องข้อผิดพลาดว่าเต็มแล้ว

เพื่อให้สามารถประมวลผลคำสั่งต่อไปนี้

MathSection m = new MathSection();

Student s1= new Student(“6011”,”John”,89);

Student s2= new Student(“6012”,”Bo”,50);

m.addNewStudent(s1); //

m.addNewStudent(s2); //

int count = m.getNumberOfStudent(); // คืนจำนวนนักเรียน คือ 2 คน

Student [] students = m.getStudents(); // คืนอะเรย์ของนักเรียนทุกคน

Student[ ] lis = new Student[2];

lis[0] = new Student(“60222”,”JJ”,60);

lis[1] = new Student(“60333”,”Koko”,90);

lis[2] = new Student(“60222”,”Jack”,60);

m.merge(lis);

double avg = m.meanOfScore(); จะคืนค่าเฉลี่ยคะแนนของนักเรียน

Student[] students = m.fillAll(); จะคืนวัตถุของนักเรียนทุกคน

m.deleteLast(); จะทำการลบข้อมูลนักเรียนคนสุดท้ายออก ทำให้เหลือแค่ “John”, “Bo”, “JJ” และ “Koko”

Student[] s = m.findByName(“J”);

จะคืนค่าเป็นอะเรย์ของตัวแปรอ้างอิงของนักเรียนที่มีชื่อที่ขึ้นต้นด้วยตัว J ได้แก่ “John”, “JJ” และ “Jack”

Student[] t = m.findByName(“K”);

จะคืนค่าเป็นอะเรย์ของตัวแปรอ้างอิงของนักเรียนที่มีชื่อที่ขึ้นต้นด้วยตัว K ได้แก่ “Koko”

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

ข้อ 3. ให้สร้าง คลาส Teacher มีฟิลด์ id, name, และ salary แทน รหัส ชื่อ คะแนน ตามลำดับ

แล้วให้สร้างคลาส Department มีอะเรย์ของครู ไม่เกิน 15 คน ถ้าเราเพิ่มเกินจะฟ้องข้อผิดพลาดว่าเต็มแล้ว

เพื่อให้สามารถประมวลผลคำสั่งต่อไปนี้

Department m = new Department();

Teacher s1= new Teacher(“6011”,”John”,8900);

Teacher s2= new Teacher (“6012”,”Bo”,50000);

m.addNewTeacher(s1); //

m.addNewTeacher (s2); //

int count = m.getNumberOfTeacher (); // คืนจำนวนนักเรียน คือ 2 คน

Teacher [] teachers= m. Teacher (); // คืนอะเรย์ของนักเรียนทุกคน

Teacher [ ] lis = new Teacher [2];

lis[0] = new Teacher (“60222”,”JJ”,60);

lis[1] = new Teacher (“60333”,”Koko”,90);

m.merge(lis);

double avg = m.meanOfSalary(); จะคืนค่าเฉลี่ยเงินเดือนของครู

Teacher t = m.getWhoMaximumSalary(); จะคืนคนที่ได้เงินเดือนสูงสุด

Teacher o = m.findByID(“6033”); จะคืน Teacher ที่มีรหัสเท่ากับ 6033 ถ้าหาไม่ได้ให้คืน null

คำแนะนำ: ให้เพิ่มฟิลด์ สำหรับนับจำนวนคน ล่าสุดเอาไว้ เพื่อจะได้รู้ว่ามีการใส่ข้อมูลมาแล้วกี่คน