

TP Stream Java

En vous basant sur les classes et objets suivants, il faudra utiliser les streams en java pour répondre aux questions, ci-après.

```
public class Student {
    private int id;
    private String name;
    private int age;
    private List<Course> courses;

    public Student(int id, String name, int age) {
        this.id = id;
        this.name = name;
        this.age = age;
        this.courses = new ArrayList<>();
    }

    public void addCourse(Course course) {
        this.courses.add(course);
    }

    // getters and setters
}

public class Course {
    private int id;
    private String name;
    private Teacher teacher;
    private int duration;

    public Course(int id, String name, Teacher teacher, int duration) {
        this.id = id;
        this.name = name;
        this.teacher = teacher;
        this.duration = duration;
    }

    // getters and setters
}

public class Teacher {
    private int id;
    private String name;
    private String department;
    private int startDate;

    public Teacher(int id, String name, String department, int startDate) {
        this.id = id;
        this.name = name;
        this.department = department;
        this.startDate = startDate;
    }

    // getters and setters
}
```

```

public static void main(String[] args) {
    // Create Teachers
    Teacher teacher1 = new Teacher("John Doe", "Mathematics", 2010);
    Teacher teacher2 = new Teacher("Jane Smith", "Physics", 2015);
    Teacher teacher3 = new Teacher("Michael Johnson", "Chemistry", 2005);
    Teacher teacher4 = new Teacher("Michel Papin", "Informatique", 2003);

    // Create Courses
    Course course1 = new Course(1, "Calculus", teacher1, 15);
    Course course2 = new Course(2, "Mechanics", teacher2, 10);
    Course course3 = new Course(3, "Organic Chemistry", teacher3, 12);
    Course course4 = new Course(4, "Java", teacher4, 12);

    // Create Students
    Student student1 = new Student("Alice", "Smith", 18);
    Student student2 = new Student("Bob", "Johnson", 20);
    Student student3 = new Student("Charlie", "Williams", 19);
    Student student4 = new Student("David", "Brown", 21);
    Student student5 = new Student("Eve", "Jones", 22);

    // Assign Courses to Students
    student1.addCourse(course1);
    student1.addCourse(course2);

    student2.addCourse(course3);
    student2.addCourse(course1);

    student3.addCourse(course4);
    student3.addCourse(course3);
    student3.addCourse(course1);

    student4.addCourse(course3);
    student4.addCourse(course1);
    student4.addCourse(course2);
    student4.addCourse(course4);

    student5.addCourse(course1);
    student5.addCourse(course3);
    student5.addCourse(course2);
}

```

- 1 - Sélectionner les Student qui ont un âge supérieur à 20 ans ?
- 2 - Transformer les Teacher en une liste de leurs noms ?
- 3 - Transformer les Teacher en une liste de leurs matières enseignées ?
- 4 - Donnez-moi la liste des teachers en passant par les courses ?
- 5 - Calculer la somme de toutes les durées des cours ?
- 6 - Sélectionner les cours avec une durée supérieure à 11 heures, puis récupérer une liste de leurs noms triée par ordre alphabétique ?
- 7 - Regrouper les Student par âge, puis afficher le nombre d'étudiants dans chaque groupe ?

8 – En passant par Student, il faut récupérer une liste de Course et il faut garder les doublons.

9 - Regrouper les Teacher par année d'entrée en fonction, puis afficher le nombre d'enseignants dans chaque groupe ? (vous pouvez utiliser une collection comme map pour récupérer le resultat).

10 - Sélectionner les Student avec un âge compris entre 18 et 20 ans, puis récupérer une liste de leurs Course triée par ordre alphabétique ?

11 - Compter le nombre d'étudiants dans chaque cours, puis récupérer une Map associant chaque Course à son nombre d'étudiants ?

12 - Quel est le nombre moyen de cours suivis par les étudiants de plus de 20 ans ?

13 - Quels sont les noms des étudiants qui ont suivi tous les cours ?