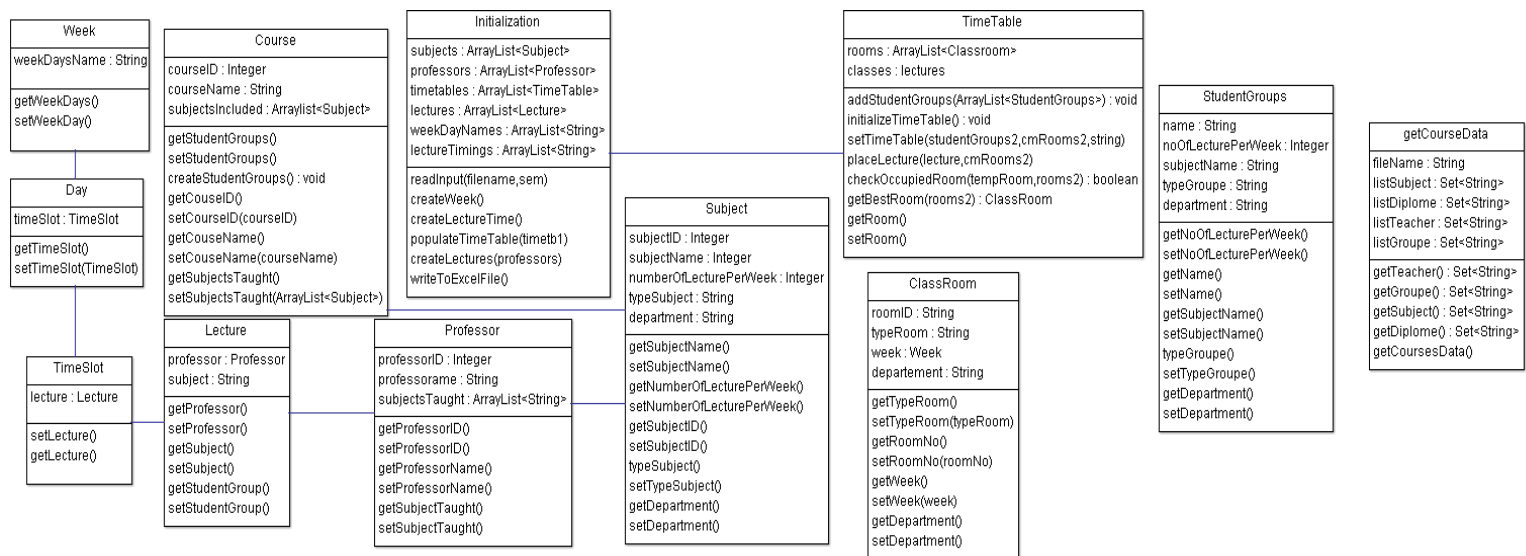


# PROJET BFA: JAVA TIMETABLE

Thanh Tu NGUYEN & Nghia Quan HOANG



## But : Faire un planning TimeTable pour le MIDO ou BFA

Attention : on veut aussi allouer des classes aux salles.

Supposer que chaque créneau de cours a lieu une heure, commencer à 9h du matin et finir à 16h du jour, de lundi à vendredi. Il y a une pause à midi pour chaque jour, et free lecture (s'il n'y a pas cours). Il n'y a que 2 salles pour 2 formations de BFA, et 15 salles pour 5 formations de MIDO. On ne considère pas de la capacité de la salle dans ce projet

Notre approche : On regroupe tous les groupes de TD (en considérant cours CM est group 0). Pour chaque group on a le groupName et matières. Pour chaque matière de chaque group on va trouver une meilleure salle disponible ainsi que meilleur TimeSlot sous quelques conditions restreintes. Sachant que 1 TimeSlot correspond à une lecture qui est associée avec un professeur et une matière (vu le graphique au-dessus pour comprendre mieux la structure)

## Etapes :

1. Reading input : On commence par récupérer tous les data depuis le fichier input (mido.csv, bfa.csv) -> Classe getCourseData
2. Classe Initialization :
  - Allouer chaque classe (BFA1, BFA2, DE1, DE2...) aux salles correspondantes de cours CM ou TD ou TP
  - Ajouter des "professors" and "subjects" (en creant classes Professor and Subject)
  - Pour chaque groupe des étudiants (nommer cours CM en group 0), on crée des « course » différents, qui sont ajoutés à timetb (class TimeTable)
  - Chercher des salles disponibles correspondantes aux cours (checkOccupiedRoom -> getBestRoom)
3. Popultating TimeTable :
  - On va randomiser tous les jours et tous les TimeSlots de chaque jour pour que les cours ont lieu à chaque jour soient au hasard (sous quelques contraintes)
4. Create lecture et timeSlot:

- Créer lecture timing (TimeSlot qui dure une heure par cours)
- Créer lecture qui est associée avec un professeur et une matière correspondante pour mettre dans timeSlot du jour.

5. Write to file csv