

1 L'objectif du projet

Le but du projet est la création d'un prototype de site web de gestion d'une université. Ce site devra gérer :

- des *personnes*, avec pour chaque : un *nom*, un *prénom* et un *identifiant* (id);
- des *parcours* (curriculums) avec pour chaque : un *nom*, un-e *secrétaire*, un-e *direct-eur-ric*e. Les parcours sont suivis par des *personnes* et contiennent des *cours*. Les élèves et les cours peuvent être partagés par plusieurs parcours, mais une personne ne devrait pas être inscrite dans deux parcours différents partageant un cours;
- des *cours* avec pour chaque : un *nom*, une *personne* qui l'enseigne, des *validations* (p. ex. un projet, des TP, des examens, une validation ne pouvant pas être partagée entre plusieurs cours); chaque cours rapporte un nombre d'*ECTS* qui dépend du parcours suivi;
- des *validations* qui ont un *nom* (p. ex. partiel, examen final, présentation), une *date*, un *coefficient* et pour chaque personne qui passe l'examen, une note;
- des *salles* qui ont un *nom* et une *capacité* d'accueil. Ces salles peuvent être occupées par des cours sur des plages horaires (c.à.d. entre une date et une heure de début et une date et heure de fin).

La moyenne d'un élève à un cours est la somme pondérée par les coefficient des points obtenus à ce cours (et si une note n'est pas dans la base, l'élève est considéré comme ayant 0). La moyenne d'un élève à un cursus est la moyenne pondérée par les ECTS.

2 Le projet

Pour ce projet, vous devez, seul-e (*mais vous pouvez, ponctuellement, vous faire aider*) :

1. Récupérer sur le moodle le squelette du projet et le faire tourner (des instructions pour Unix sont données dans le fichier README).
2. Créer une base de données SQLite dans le fichier `univ.db` avec un schéma de votre choix. Vous êtes libres de la forme du schéma mais il doit permettre de modéliser ce qui est décrit dans la section 1. Vous pouvez cependant ajouter tous les contraintes d'intégrité qui vous semblent pertinentes.
3. Les fonctions dans le fichier `models.py` correspondent à des requêtes SQL (dont la fonctionnalité est décrite dans les commentaires). Vous devez écrire les requêtes (les *TODO*) et que le site marche correctement.

Cependant, si vous avez complété les tâches ci-haut, vous êtes invité à améliorer les fonctionnalités du site (ça implique donc de modifier le code Python). Vous êtes tout à fait libre sur cette partie mais vous pouvez, par exemple :

- gérer les erreurs d'intégrité en s'inspirant de ce qui a été fait dans le fichier `run.py` pour l'ajout d'une nouvelle note (dans la fonction `showValidation`);
- améliorer le style général du site web;
- ajouter les fonctions manquantes (suppression de notes, etc.).

3 Rendu

Ce projet doit être rendu avant le 23 avril 23:59 sur le moodle par un fichier *zip* ou *tar.gz* contenant :

1. votre code modifié;
2. un fichier `schema.sql` qui crée une base de données au schéma attendu par votre code et qui peuple cette base de données avec des personnes, des cours, des salles, etc. pour pouvoir tester immédiatement le site;
3. Un fichier PDF expliquant ce que vous avez fait ou non dans le projet et les choix que vous avez effectué (quelles contraintes d'intégrités ? comment sont-elles implémentées ? quelles extensions ? etc.).

4 Conseils

Commencez rapidement ! Cela vous permettra de vérifier que vous arrivez à faire tourner le projet ainsi qu'à modifier le site (et poser toutes les questions utiles pendant les TP/TD si vous n'y arrivez pas). Commencez par créer le fichier `schema.sql` afin de pouvoir rapidement manipuler le site. Les fonctions à remplir dans le fichier `models.py` sont plus ou moins triées par onglet qu'elles débloquent ; testez au fur et à mesure que tout marche bien. Faites des sauvegardes régulières (utilisez *git* si vous connaissez). Vous pouvez soumettre plusieurs fois, n'attendez pas le dernier jour !