

Évaluation de Travaux Pratiques

avril 2023
durée : 1h
responsable : M.Munier

Sujet 2 - Étudiants & notes

Documents autorisés : Pour cette évaluation de travaux pratiques vous êtes autorisés à utiliser tous les documents pédagogiques qui vous ont été fournis : support de cours "analyse et traitement de données structurées" (module R208) (éventuellement annotés) distribué par M.Munier, corrections des TD/TP, notes de TD et notes de TP. Tout autre document (documentations récupérées sur le web, photocopies de livres, etc...) est formellement interdit. Dans le même ordre d'idée, aucune assistance électronique (calculatrice, ordinateur de poche, smartphone, mail, Google+, Facebook, Twitter, Google Glass, ChatGPT,...) n'est autorisée. Toutes les notions vues en cours sont supposées acquises et, le cas échéant, doivent être mises en œuvre.

Restitution de vos travaux : Vous développerez toutes les classes de cette évaluation dans un seul et unique répertoire (éventuellement un seul fichier) dont le nom est **votre nom de famille en minuscules**. À la fin de la séance votre "copie d'examen" sera une archive (formats **zip** ou **tar** uniquement) contenant ce répertoire et dont le nom sera votre nom de famille en minuscules, sans accents et sans espaces. Vous enverrez ensuite votre archive par mail à l'adresse manuel.munier@gmail.com

Conseils : Durant cette évaluation de TP (qui ne dure que 1h), votre but n'est pas de tout faire, mais plutôt de nous prouver que vous êtes capables de vous débrouiller en programmation objet en Python devant une machine. Cela signifie bien évidemment savoir écrire des classes correctes (constructeurs, méthodes, attributs,...), mais également savoir interpréter les erreurs de syntaxe et/ou d'exécution, les corriger, tester vos classes, etc... Dit de manière un peu plus directe, **il est plus "intéressant" pour vous d'avoir un programme fonctionnel validant une partie des classes que d'avoir écrit toutes les classes mais sans en avoir testé aucune !**

Tout manquement sur la procédure de restitution de votre travail sera immédiatement sanctionné sur votre note !

★ ★ ★

Dans cet exercice on se propose de programmer en Python un ensemble de fonctions nécessaires à la réalisation d'une application de gestion des étudiants et de leurs notes. Je vous rassure tout de suite, en ce qui concerne les notes nous nous limiterons à une série de notes pour chaque étudiant sans coefficient et sans UV. Ça nous (ou plutôt **vous**...) simplifiera le calcul des moyennes.

1. Un **Etudiant** sera un tuple qui représentera un étudiant avec son nom et son prénom (chaînes de caractères), son âge (un entier) et la liste de ses notes (un tableau de réels). Écrire les fonctions réalisant les opérations suivantes :
 - de créer un nouvel étudiant à partir de son nom, de son prénom et de son âge ; son tableau de notes sera créé vide ; cette fonction **creerEtudiant(string,string,réel)** retournera un tuple de type **Etudiant**
 - d'enregistrer une nouvelle note (fonction **enregistrerNote(Etudiant,réel)**), sachant qu'au départ (i.e. à la création) l'étudiant ne dispose d'aucune note
 - d'afficher les infos d'un étudiant (fonction **afficherEtudiant(Etudiant)**)
 - de retourner le nombre de notes enregistrées (fonction **nbNotes(Etudiant)**)
 - de calculer la moyenne des notes enregistrées pour cet étudiant, sans coefficient ni UV (fonction **calculerMoyenne(Etudiant)**). NB : si l'étudiant en question ne dispose d'aucune note, cette fonction retournera 0
2. Une **Promotion** sera simplement un tableau qui représentera une promotion particulière avec la liste de tous les étudiants de cette promotion. À la construction d'une promotion, il n'y a bien sûr aucun étudiant. Vous devrez implémenter les fonctions suivantes :
 - d'ajouter un étudiant à la promotion ciblée (fonction **inscrire(Promotion,Etudiant)**)
 - de connaître le nombre d'étudiants inscrits (fonction **nbEtudiants(Promotion)**)
 - de calculer la moyenne de la promotion (fonction **calculerMoyenne(Promotion)**) en excluant bien évidemment les étudiants qui ne disposeraient encore d'aucune note
3. Afin de pouvoir valider votre travail, il vous est naturellement demandé d'écrire un programme de test créant plusieurs étudiants, de les ajouter à une promotion, et d'invoquer les différentes fonctions. NB : Je vous rappelle que vos fonctions, sauf mention contraire, ne doivent pas faire de lecture ni d'affichage. Les entrées/sorties avec l'utilisateur se font uniquement dans le programme principal.

★ ★ ★