

# Examen #2 pratique

## Énoncé :

Développez une application java avec une interface utilisateur graphique (IUG) qui permet de définir une classe **Professeur** et manipuler un **ArrayList listeProfs** de professeurs.

## Détails des classes

### ✓ Classe **Professeur**

Un professeur est caractérisé par les attributs d'instance suivants: son **nom** et ses **deux taches (Automne, Hiver)** et un attribut de classe (static) : **nombre de professeurs**.

Voici la liste des méthodes que la classe doit avoir:

- un constructeur par défaut
- un constructeur qui prend en paramètre un nom et deux taches (utilisant les méthodes **set**);
- les méthodes **set ( )** pour les 3 attributs d'instances en s'assurant que les **taches** transmises sont **entre 0 et 50** sinon **mettre la tache 0** ;
- les méthodes **get ( )** pour les 3 attributs.
- la méthode **calculerTache ( )** : pour calculer sa tache annuelle;
- la méthode **toString ( )** : pour afficher ses détails ;
- la méthode **message ( )** : qui retourne **plein** si sa tache annuelle est supérieure ou égale à **88** et **partiel** autrement.

### ✓ Classe **FenetreDev** avec une IUG de votre choix permettant de (ce sont de méthodes de la classe ceux en rouge) :

- **créer** le **ArrayList listeProfs** ;
- **initialiser** **listeProfs** avec 3 professeurs;
- **afficher** la liste des 3 professeurs
- **ajouter** d'autres profs
- **calculer** les statistiques suivantes du groupe de professeurs;
  - la moyenne de tache,
  - la tache la plus basse,
  - la tache la plus haute ;
- **modifier** les taches des 2 professeurs suivants:
  - le **1<sup>er</sup>** professeur :
    - sélectionner le **1<sup>er</sup>** professeur ;
    - afficher ses informations ;
    - remplacer sa tache Automne par **48** ;
  - le **3<sup>ème</sup>** professeur:
    - sélectionner le **3<sup>ème</sup>** professeur ;
    - afficher les informations de cet professeur;
    - remplacer sa tache Hiver par **42**

## Examen #2 pratique

Ceci est un exemple d'interface. Vous devez créer votre propre style d'interface.

Voir l'exécution ci-après

*au clic sur le bouton Ajouter*

**Gestion Professeurs**

Nom: Robert Automne: 0 Hiver: 38

Liste position: 0

Ajouter Lister Effacer Quitter

Modifier Stats du groupe

Résultats

Fait partie de l'interface

*au clic sur le bouton Lister*

**Gestion Professeurs**

Nom: Alain Automne: 48 Hiver: 48

Liste position: 2

Ajouter Lister Effacer Quitter

Stats du groupe

Résultats

Nom	Automne	Hiver	Année	Temps
Robert	0	38	38	partiel
Sylvie	0	22	22	partiel
Alain	48	48	96	plein

*au clic sur le bouton Stats du groupe*

**Gestion Professeurs**

Nom: Alain Automne: 48 Hiver: 48

Liste position: 2

Ajouter Lister Effacer Quitter

Stats du groupe

Résultats

Moyenne de la tache : 52  
Tache la plus basse : 22  
Tache la plus haute : 96

*au clic sur le bouton Lister après avoir  
modifié la tache Automne de Robert par 50  
et la tache Hiver de Alain par 42*

**Gestion Professeurs**

Nom: Alain Automne: 48 Hiver: 42

Liste position: 2

Ajouter Lister Effacer Quitter

Stats du groupe

Résultats

Nom	Automne	Hiver	Année	Temps
Robert	50	38	88	plein
Sylvie	0	22	22	partiel
Alain	48	42	90	plein

## Examen #2 pratique

### Évaluation

Qualité de(s) l'interface(s) graphique(s)	25%
Les 5 opérations	60% (12% par opération)
Vidéo des tests	10%
Qualité de l'affichage des résultats	5%

### Remise

Le projet sera déposé dans LEA. Lors de la remise placez l'url de votre vidéo des tests. Aucun fichier vidéo, seulement le lien de la vidéo sur Vimeo, YouTube ou autre. Dans votre vidéo ne pas montrer du code seulement l'exécution. Si un exercice ne fonctionne pas, vous pouvez alors montrer le code de cette option et expliquez selon vous pourquoi cela n'a pas fonctionné.

**Pour les tests suivre l'ordre des opérations. Après chaque changement cliquer sur «Lister»**

**Toute consigne non respectée vous fera perdre 10% de la note finale.**