**TP3 : interactions entre objets**

1. **Projet mail-system**
2. Ouvrir le projet mail-system. Ce projet ne contient que les exécutables des classes, mais pas le code source. Vous pouvez donc uniquement créer des instances des classes MailItem, MailClient, et MailServer et examiner leur contenu.
3. Soit le scénario suivant :

*Camille et Cléo possèdent un client de messagerie, et partagent le même serveur de mail. Camille envoie un message à Cléo avec comme texte : “Bonjour !” Cléo affiche sur son terminal le message envoyé par Camille.*

Simulez ce scénario avec *BlueJ* en utilisant les trois classes à disposition, et listez la suite d’opérations effectuées.

|  |
| --- |
|  |

1. Complétez le tableau suivant pour faire apparaître les attributs avec leur type et les en-têtes des méthodes, incluant les constructeurs de la classe MailItem, MailClient et MailServer. Pour chaque méthode ajoutez un commentaire pour expliquer son fonctionnement.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MailItem** | **MailClient** | **MailServer** |
| **Attributs** |  |  |  |
| **Constructeurs** |  |  |  |
| **Autres méthodes** |  |  |  |

1. Écrivez maintenant le code source de la classe MailItem. Il faudra d’abord supprimer la classe existante, puis en créez une nouvelle.
2. Écrivez maintenant le code source de la classe MailClient.
3. Vérifiez votre implémentation : bon fonctionnement ? code lisible ? code commenté et indenté ?
4. **L’horloge numérique**
5. Ouvrir le projet clock-display.
6. Créez un objet de la classe ClockDisplay en sélectionnant le constructeur new ClockDisplay().
7. Quels sont les attributs (et leur type) de la classe ClockDisplay ? Quelles sont les méthodes (en incluant les constructeurs) ?

|  |
| --- |
|  |

1. Créez un objet de la classe NumberDisplay. Quels sont les attributs (et leur type) de cette classe ? Quelles sont les méthodes (en incluant les constructeurs) ? Expliquer le fonctionnement général de la classe. Supprimez ensuite cet objet.

|  |
| --- |
|  |

1. Combien de fois faut il invoquer la méthode timeTick pour que l’heure affichée soit 01:00 ?

|  |
| --- |
|  |

1. Comment le même résultat peut il être obtenu plus rapidement ? Donnez deux solutions possibles.

|  |
| --- |
|  |

1. Modifiez la classe ClockDisplay pour qu’elle soit capable de gérer et d’afficher des secondes. Vous modifierez notamment la méthode timeTick pour incrémenter les secondes, et éventuellement les minutes et les heures.
2. Ajoutez un troisième constructeur à la classe ClockDisplay qui initialise l’heure en fonction d’un temps donné en seconde. Par exemple,

|  |
| --- |
| ClockDisplay cd = new ClockDisplay(3993); |

doit créer une horloge qui a pour horaire : 01:06:33.

1. **Pour aller plus loin [facultatif]**

Reprenez le projet de l’horloge numérique. Modifiez l’implémentation pour avoir un affichage sur 12 heures (comme dans les pays anglo-saxons où 16h23 est affiché à 4:23 pm). Pour information : après minuit et midi, l’affichage est 12:00, et non 00:00. Les heures vont donc de 1 à 12, et non pas de 0 à 11.

Il y a (au moins) deux solutions pour réaliser cet affichage. Implémentez les.

Quelle solution est la plus facile ? Quelle est la meilleure ?

|  |
| --- |
|  |