

## Travail pratique 2-3

**Voir à la fin de cet énoncé pour l'évaluation et remise de ce labo.**

### Objectifs

- Composition d'objets
- Insertion d'un objet dans un tableau d'objets
- Retrait d'un objet dans un tableau d'objets
- Utilisation de méthodes privées
- Utilisation de `JOptionPane.showInputDialog` et de `JOptionPane.showMessageDialog`

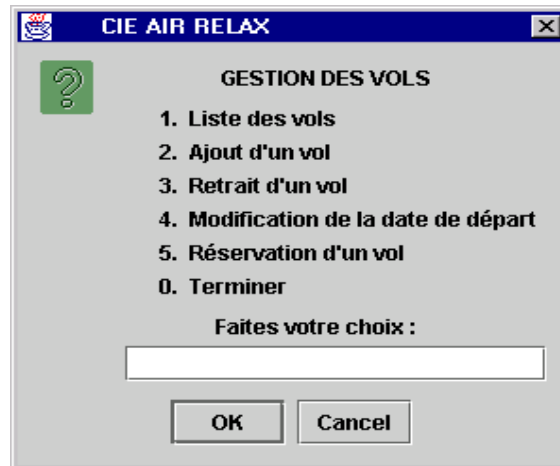
### Sujet

On vous engage afin de réaliser un projet de gestion des vols d'une compagnie aérienne. Votre projet sera utilisé par différentes compagnies aériennes. Chaque compagnie dispose d'un fichier (des vols déjà existants) dont **le nom est celui du nom de la compagnie suivie de l'extension .txt**. Chaque enregistrement des fichiers contient :

- Le numéro du vol
- La destination du vol
- La date de départ du vol
- Le nombre total des réservations à date

Les enregistrements sont par ordre croissant du numéro du vol. **On ne connaît pas le nombre d'enregistrements qu'il y a dans les fichiers.**

Dans votre projet, la classe qui contiendra le « main » devra faire afficher un menu : par exemple, pour la compagnie « Cie Air Relax » nous aurons le menu suivant, à l'aide de `JOptionPane.showInputDialog`



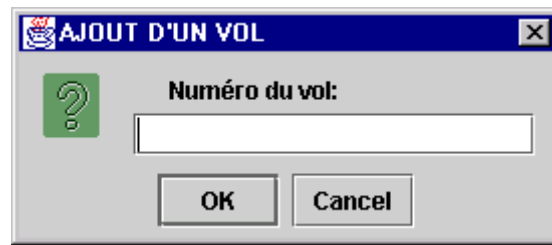
N'oubliez pas de produire un message d'erreur si le choix n'est pas entre 0 et 5

### Travail à faire

Votre projet va comporter 4 classes soient :

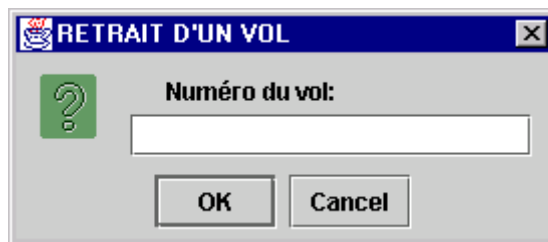
1. La classe **Date** ayant les attributs entiers **jour**, **mois**, **an**, le constructeur par défaut et celui avec 3 paramètres, les méthodes « **get** » et la méthode **toString** qui retourne la chaîne contenant les attributs séparés par "/" et en plus les attributs jour et mois dans le format décimal "00"
2. La classe **Vol** dont les caractéristiques sont :
  - attributs **d'instances** : numéro du vol, sa destination, la date de départ (objet de la classe Date), et le nombre total de réservations à date
  - le constructeur permettant d'initialiser les 4 attributs d'instances avec les valeurs reçues en paramètres
  - les méthodes « **set** » pour seulement le total des réservations et la date de départ
  - les méthodes « **get** » pour tous les attributs
  - la méthode **toString()** qui retourne une chaîne contenant les attributs séparés par des "\t"
3. La classe **Compagnie** dont les caractéristiques sont :
  - la constante de classe publique **MAX\_PLACES = 340** (le nombre de places que contient un avion)
  - attributs d'instances :
    - le nom de la compagnie,
    - le **nombre maximum** de vols que peut avoir la compagnie,
    - un tableau d'objets de la classe Vol (**on ne fait que déclarer la référence au tableau**)
    - le nombre de vols actifs (un compteur)
  - le constructeur à 2 paramètres qui permet d'initialiser le nom de la compagnie et le nombre maximum de vols avec les valeurs reçues en paramètres. De plus, le constructeur va réserver l'espace mémoire **maximum** pour le tableau et ensuite faire l'ouverture et la lecture du fichier (Cie Air Relax.txt) situé dans **Studium** de façon à remplir le tableau d'objets Vol et de compter les vols. Ensuite, il ferme le fichier.

- La méthode d'instance **insérerVol** doit s'assurer qu'il reste de la place dans le tableau des vols, sinon prévoir un message d'erreur adéquat avec `JOptionPane.showMessageDialog`. S'il reste de la place, elle permet de saisir le numéro du nouveau vol à l'aide de `JOptionPane.showInputDialog` sous la forme :



Elle fera appel à la **rechercherVol** (une méthode privée de la classe Compagnie) qui reçoit en paramètre le numéro du vol, car pour pouvoir ajouter un nouveau vol, le numéro du vol ne doit pas déjà exister (prévoir un message d'erreur approprié si ce n'est pas le cas). Si le numéro est accepté, saisir alors la destination, la date de départ (`JOptionPane.showInputDialog`), mettre à 0 le total des réservations et insérer le vol au bon endroit dans le tableau.

- La méthode d'instance **retirerVol** permet de saisir le numéro du vol à l'aide `JOptionPane.showInputDialog` sous la forme :




Elle fera appel à **rechercherVol** car pour pouvoir retirer un vol, le numéro du vol doit être existant (prévoir un message d'erreur approprié si ce n'est pas le cas). Si le numéro existe, à l'aide d'une seule fenêtre `JOptionPane.showInputDialog`, faire afficher sa destination, sa date et le total des réservations et demander la confirmation du retrait : « **Désirez-vous vraiment retirer ce vol (O/N) ?** » Si oui, retirer le vol du tableau en l'écrasant et ne pas oublier de diminuer le nombre de vols actifs.

- La méthode d'instance **modifierDate** permet de saisir le numéro du vol dont on veut modifier la date, à l'aide de `JOptionPane.showInputDialog` (la barre de titre contiendra le titre **MODIFICATION DE LA DATE DE DÉPART**). Elle fera appel à **rechercherVol**. Si le numéro n'existe pas prévoir le message d'erreur approprié, sinon à l'aide d'une seule fenêtre `JOptionPane.showInputDialog` faire afficher sa destination, sa date de départ actuelle et demander l'entrée de la nouvelle date (dans la forme JJ/MM/AAAA).
- La méthode d'instance **reserverVol** permet de saisir le numéro du vol que le client désire réserver, à l'aide de `JOptionPane.showInputDialog` (la barre de titre contiendra le titre **RÉSERVATION D'UN VOL**). Elle fera appel à **rechercherVol**. Si le numéro n'existe pas prévoir le message d'erreur approprié, **sinon s'il reste de la place dans l'avion, à l'aide d'une seule fenêtre `JOptionPane.showInputDialog` faire afficher sa destination, sa date de départ, le nombre de places restantes dans l'avion** et demander l'entrée du nombre de places que le client désire

réserver et faites la réservation. S'il n'y a plus de place, faire afficher à l'aide de `JOptionPane.showMessageDialog` le message adéquat.

- La méthode d'instance **listeVols** qui fait afficher la liste des vols existants à l'aide de `JTextArea` [fonte courier, de style `Font.PLAIN` et de taille 12] à transmettre à `JOptionPane.showMessageDialog`, sous la forme (la barre de titre contiendra le nom de la compagnie):



Numéro	destination	Date départ	Réservations
14567	Toronto	12/04/2002	167
17890	Guadeloupe	01/05/2002	340
19012	Nouvelle-Orléans	03/05/2002	250
22345	Martinique	25/04/2002	337
28901	Las Vegas	13/05/2002	0
36789	New-York	15/04/2002	67
45678	Les Iles Baléares	06/05/2002	150
53456	San Francisco	13/04/2002	327

- La méthode d'instance **ecrireFichier** qui ouvre le fichier de la compagnie en écriture et écrit chaque objet du tableau dans le fichier selon le format **19012;Nouvelle-Orléans;03;07;2021;250** (ceci est un exemple, par exemple l'objet dans le tableau à la position 2).
  - La méthode privée **insere( )** dont le seul paramètre est un objet Vol (celui à insérer)
4. La classe **GestionCompagnie** qui contient le « main » dont l'objectif est de créer un objet de la classe **Compagnie** dont le nom doit être fourni par l'utilisateur en utilisant une fenêtre `JOptionPane.showInputDialog` (vous devez taper **Cie Air Relax**) et dont le nombre maximum de vols possible doit être fourni par l'utilisateur en utilisant une fenêtre `JOptionPane.showInputDialog` (vous devez taper **15**). Par la suite, le menu doit s'afficher tant et aussi longtemps que l'utilisateur n'aura pas fait le choix 0. Lorsqu'il fait le choix 0, on doit écrire le fichier.

**NOTA** : Ce travail peut être fait par des équipes de maximum trois étudiants.

Déposer votre travail compressé dans LEA.

## Critères de correction

-50% si rien est fonctionnel et le code contient des erreurs

Qualité du code et utilisation de la POO et respect des consignes du labo	10%
Fonctionnalités du menu, il y en a 8 x1 10% chaque	80%
Utilisation et gestion du menu, validation, indentation du code, commentaires	5%
Ergonomie (présentation des résultats et écran de saisie de données)	5%

## REMISE :

Faites des copies écran de : votre menu, résultat de l'option1, option 2, option 3, option 4 et option 5. Ne pas oublier d'afficher aussi ce que vous avez tapé. Copiez ces images dans un document Word nommé « **labo2-3Résultats** ». Indiquez dans ce même document *toute question qui ne fonctionne pas et donner une idée du provient le problème*. Je regarderai le code pour pouvoir vous donner quelques points au lieu de zéro. Si **pas de copie écran -50%** de la question. Si **une question ne fonctionne pas** et vous n'avez pas indiqué dans votre rapport alors vous aurez zéro pour la question.

Respectez ces règles, ce sera compliqué après pour apporter des correctifs à vos notes. Les excuses du non-respect de ces consignes ne seront pas considérées.

**Déposez « Labo2-3 » dans le dossier « documents » absolument. Les pénalités seront de -20% si cette consigne n'est pas respectée et -10% pour toute autre consigne non respectée donnée dans cet énoncé.**