

Examen Techniques et Outils de Développement

8 janvier 2019

Stéphane Lopes, Zoubida Kedad

Durée 2h – Documents non autorisés

Exercice 1 - tests unitaires avec JUnit

Dans cet exercice, on veut réaliser une classe `Montant` représentant une somme dans une certaine devise.

Un montant comporte une *somme* (de type réel) et une *devise* (de type chaîne). On suppose que les *getters* existent et fonctionnent correctement. Un montant ne peut pas être négatif.

Pour chaque cas de test, vous donnerez le code de la méthode de test ainsi que l'implémentation de la méthode testée.

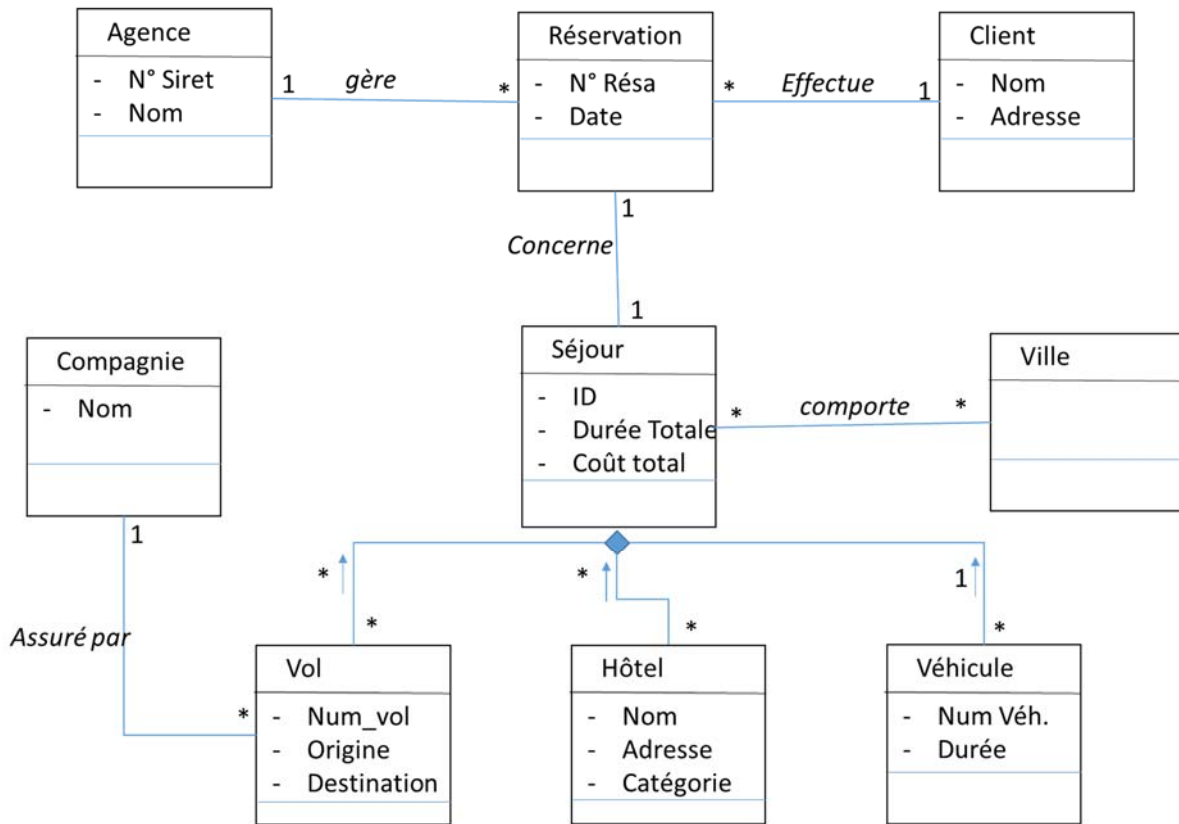
1. [Cas de test] Construction d'un montant de 1.5€.
2. [Cas de test] Pour un montant de 1.5€, la méthode `toString()` retourne "1.5€".
3. Qu'est qu'un *test fixture* ? Ajoutez-le à la classe de test pour simplifier les tests précédents.
4. [Cas de test] Un montant de 1.5€ est converti en un nouveau montant de 1.725\$ par la méthode `convertTo("$")` (taux de change 1€ = 1.15\$).
5. [Cas de test] Construction d'un montant de -1.5€ (lance une exception de type `IllegalArgumentException`).

Exercice 2 – UML Diagramme de classes

Le diagramme de classes (partiel) donné ci-après décrit des réservations de séjour faites par des clients auprès d'agences de voyage. Trois éléments peuvent entrer dans la composition d'un séjour : des vols, des hôtels et des locations de véhicules.

Selon les informations figurant dans le diagramme, dites si les affirmations suivantes sont vérifiées ou non en justifiant vos réponses.

1. Un vol peut entrer dans la composition de deux séjours distincts.
2. Un même client peut réserver plusieurs vols pour un même séjour.
3. Une réservation peut concerner plusieurs compagnies aériennes.
4. A un séjour correspond un client unique.
5. Une location de véhicule ne peut être faite que pour un seul client.
6. Un vol correspond à un seul client.



Exercice 3 – UML Diagramme de classes

Compléter le diagramme de classe donné à l'exercice 2 pour prendre en compte les informations suivantes :

1. On souhaite retrouver la ville dans laquelle chaque compagnie aérienne est localisée.
2. On souhaite pouvoir calculer le coût total d'un séjour, en additionnant les prix de chacun de ses composants.
3. Les locations de véhicules sont faites auprès d'entreprises de location, chacune étant localisée dans une ville.
4. Chaque réservation fait l'objet d'un paiement par le client. On souhaite disposer du montant et de la date du paiement. Un client ne peut pas régler une réservation qu'il n'a pas effectuée.

Vous pouvez faire l'exercice directement sur la feuille d'énoncé.

Exercice 4 – UML Diagramme de séquence

En considérant le diagramme de classes obtenu à l'exercice 3, donner les diagrammes de séquences correspondant aux deux scénarios suivants :

1. Réservation par le client C1 du séjour S1 qui comporte les éléments suivants : départ de Paris destination Rome par le vol V1, départ de Rome destination Florence par le vol V2, puis retour vers Paris par le vol V3. Le client souhaite séjourner à l'hôtel H1 à Rome et à l'hôtel H2 à Florence. Il ne souhaite pas disposer d'un véhicule pendant son séjour.
2. Règlement du séjour S1 (décrit ci-dessus) par le client C1.

Exercice 5 – UML Diagramme d'activités

On suppose que le règlement d'un séjour par le client s'effectue en ligne. Le client doit d'abord s'authentifier s'il possède un compte, ou en créer un s'il n'en possède pas. Une fois connecté, il peut demander à procéder au règlement de sa réservation. Il doit fournir le numéro de réservation afin que le système vérifie qu'il s'agit bien de la sienne. Le prix total du séjour est alors calculé, en recherchant les vols et les hôtels de sa réservation et en sommant les montants correspondants. Ensuite, les locations de véhicules éventuelles sont ajoutées. Le montant total est alors transmis au client, qui saisit ses coordonnées bancaires pour le paiement. Une fois le paiement enregistré, une facture est envoyée au client.

Donner le diagramme d'activités correspondant au processus de règlement.