

Demande initiale

Votre client vous demande de lui livrer ces fonctionnalités pour le 16 février 2015, avant 17h30. La date de livraison n'est pas négociable.

L'application à développer est un logiciel qui calculera des remboursements de réclamations d'assurances de soins de santé.

Dans ce domaine, il existe beaucoup de clauses, de cas d'exceptions et de particularités avec les contrats d'assurance, et les contrats de notre client ne font pas exception. Il a donc besoin d'un coup de main pour l'aider à se retrouver dans toutes ses règles afin d'assurer une croissance à son entreprise.

Le logiciel ne possèdera pas d'interface utilisateur car il est destiné à être invoqué à partir d'une application web. Le contrat ne consiste donc qu'au développement du "back-end" de l'application.

Fonctionnalités

Le fichier d'entrée, en format JSON, aura l'air de ceci :

```
{
  "client": "100323",
  "contrat": "A",
  "mois": "2017-01",
  "reclamations": [
    {
      "soin": 100,
      "date": "2017-01-11",
      "montant": "234.00$"
    },
    {
      "soin": 200,
      "date": "2017-01-13",
      "montant": "90.00$"
    },
    {
      "soin": 334,
      "date": "2017-01-23",
      "montant": "125.00$"
    }
  ]
}
```

Le fichier de résultat généré par le logiciel devra ressembler à ceci :

```
{
  "client": "100323",
  "mois": "2017-01",
  "remboursements": [
    {
      "soin": 100,
      "date": "2017-01-11",
      "montant": "58.50$"
    },
    {
      "soin": 200,
      "date": "2017-01-13",
      "montant": "22.50$"
    },
    {
      "soin": 334,
      "date": "2017-01-23",
      "montant": "0.00$"
    }
  ]
}
```

Le programme devra prendre le fichier d'entrée comme argument lors de l'exécution du logiciel dans une console (paramètre au main). Le fichier de sortie devra également être spécifié à la console. Exemple :

```
java -jar Refund.jar inputfile.json refunds.json
```

Il existe 4 types de contrat différents : A, B, C, D. Voici les différentes catégories de soin en fonction de leur numéro et les remboursements pour chaque type de contrat :

Número de soin	Catégorie de soin	A	B	C	D
0	Massothérapie	25%	50% max 40\$	90%	100% max 85\$
100	Ostéopathie	25%	50% max 50\$	90%	100% max 75\$
200	Psychologie individuelle	25%	100% max 70\$	90%	100% max 100\$
[300..399]	Soins dentaires	0%	50%	90%	100%
400	Naturopathie, acuponcture	0%	0%	90%	100% max 65\$
500	Chiropratie	25%	50% max 50\$	90%	100% max 75\$
600	Physiothérapie	40%	100%	90%	100% max 100\$
700	Orthophonie, ergothérapie	0%	70%	90%	100% max 90\$

Voici également quelques validations à effectuer sur les données d'entrée :

- Le numéro de client doit obligatoirement être composé de 6 chiffres, aucun autre caractère que des chiffres ne doit se retrouver dans le numéro de client.
- Le contrat doit être une des quatre lettres suivantes : A, B, C, D. La lettre doit toujours être en majuscule.
- L'application recevra toutes les réclamations d'un client pour un mois dans le même document JSON, le mois est spécifié dans le champs mois sous le format AAAA-MM.
- Il faut vérifier que chacune des réclamations est bel et bien faite pour le mois qui est traité. Un soin fait le 2 février 2017 ne doit pas être considéré dans une feuille de janvier 2017. La

date de consommation de chaque soin est spécifiée sous le format ISO 8601 (AAAA-MM-JJ).

- Le signe de dollar (\$) est toujours présent à la fin d'un montant.
- Le numéro de soin doit être valide selon le tableau ci-haut.

Les données d'entrée doivent respecter toutes ces règles pour être considérées valides. Si les données d'entrée ne sont pas valides, l'application devra générer le fichier de sortie suivant :

```
{  
  "message": "Données invalides"  
}
```

Contraintes technologiques

Voici les contraintes que vous devez respecter :

- Le logiciel doit être développé avec le langage de programmation Java (JDK8).
- Il est impératif d'utiliser l'environnement de développement intégré NetBeans 8.0 ou plus.
- Les fichiers d'entrées et de sorties doivent être des documents JSON.
- Les sources doivent être entreposées dans un dépôt GIT sous github.