

Enoncé

Rendez-vous sur <https://classroom.github.com/a/hAxACtb->, téléchargez votre projet, lisez attentivement la section **Scénario** et réalisez le travail demandé dans la section **Exercices**. Un fonctionnement correct ne peut pas vous assurer une note parfaite. La qualité de votre code, performance et lisibilité sont également pris en compte ! Pensez également à commenter votre code si vous le jugez complexe.

Pour finir, n'oubliez pas de *pusher* votre code sur votre *repository* git – vous avez jusqu'au 17 novembre 2019 à 23h59.

Scénario

Vous êtes mandatés par une entreprise de marketing qui a publié un jeu sur un réseau social donné. Ce jeu permettait de gagner un cadeau et d'inviter d'autres personnes à jouer au jeu.

Cette entreprise vous demande de créer un arbre et de leur fournir certaines méthodes leur permettant de faire des statistiques concernant les participants du jeu.

Pour cela, deux fichiers vous sont fournis :

- customers_data.csv
- customers_relations.csv

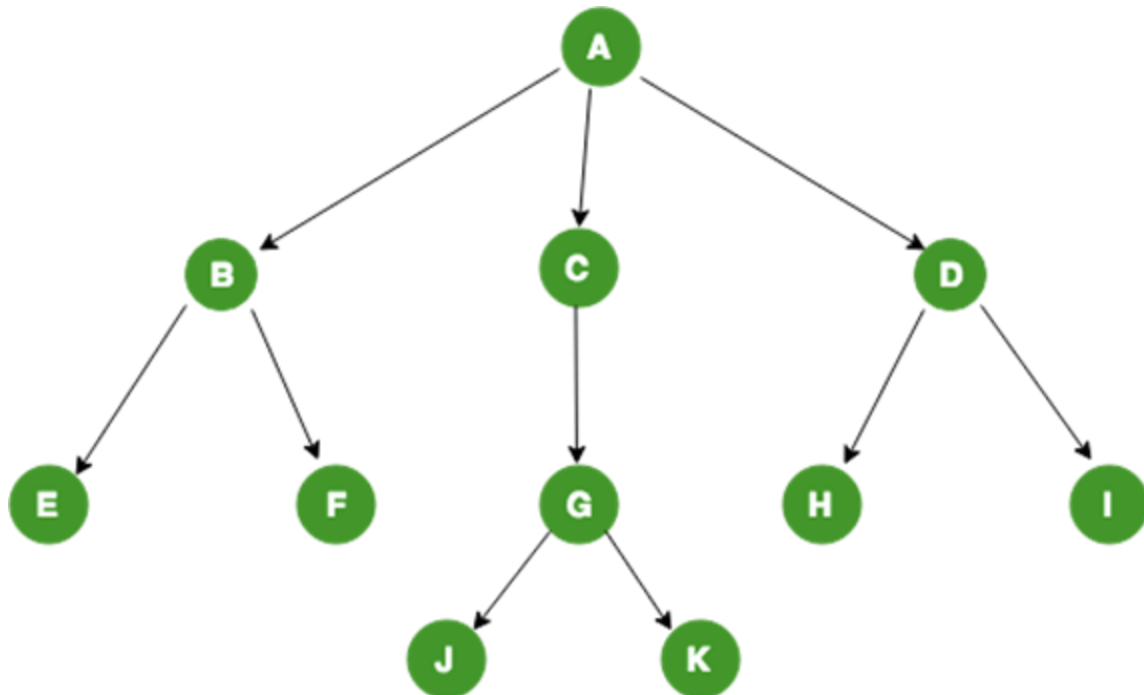
Dans le premier fichier, vous avez chaque client ayant participé au jeu. Dans le deuxième, vous avez les liens entre les clients et ces liens représentent les invitations.

Le fichier contenant les invitations est structuré de la façon suivante :

```
id, invitation
1,2
1,3
1,4
1,7
...
```

Chaque client étant identifié par le champ « id », nous pouvons voir que le client numéro « 1 » a invité le client numéro 2, 3, 4, 7, etc.

Après avoir combiné les deux fichiers en un seul arbre, vous devriez avoir la structure suivante :



Chaque lettre correspond à un client. Aussi, nous devons considérer l'élément A comme étant à la profondeur 0 – B, C, D à la profondeur 1, et ainsi de suite.

Exercice

Dans cet exercice, vous avez 4 classes à coder, réparties dans 3 packages différents. Voici la structure finale de votre projet :

- domaine
 - customer
- metier
 - customers
- outils
 - file_loader
 - file_parser

Les fichiers se trouvent dans un dossier data.

Codez les méthodes se trouvant dans les fichiers ci-dessus en vous basant sur les commentaires qui vous sont donnés. Vous pouvez créer autant de méthodes et de classes que vous le souhaitez mais vous devrez aussi les commenter – **la pertinence des commentaires sera aussi évaluée !**

Pour finir, vous devez laisser la possibilité à l'entreprise de pouvoir afficher le contenu de l'arbre.

Pour cela, lorsque vous faites un *print* de votre arbre, voici comment vous devrez l'afficher :

```
Customer Remy Reese
|__Customer Allison Bingham
||__Customer Julius Morrison
|||__Customer Alba Ripley
....
|__Customer Barney Curtis
||__Customer Cristal Gardner
|||__Customer Abdul Archer
||||__Customer Barry Ranks
||||_Customer ...
```

Si votre output est trop grand, vous avez une méthode à disposition dans `file_writer` vous permettant d'enregistrer votre string dans un fichier de texte.