

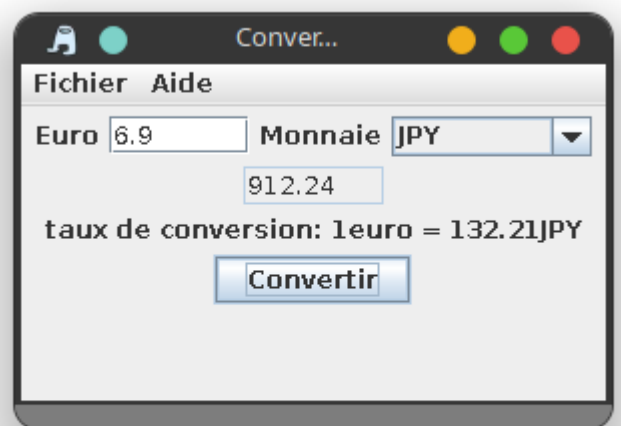
LE2 - Programmation Java: TP3/4

Tous les codes sources et la javadoc sont déjà disponibles sur [le dépôt GitHub](#).

Convertisseur 2.0:

Par rapport à la version 1.0 le CHANGELOG du convertisseur graphique de monnaie correspond à:

- Ajout d'un dropdown menu pour sélectionner la devise résultante (maintenant supporté USD/JPY/GBP/MAD)
- Taux de conversion mis à jour FOREX 25/06/2021 @ 13:25 UTC
- Bugs d'affichage: correctifs
- nouvel exemple de conversion:



Convertisseur 3.0:

Par rapport à la version précédente, les taux de change sont actualisés via le site web [Change Magenta](#).

La Javadoc a également été mise à jour; disponible ici:

<https://perso.esiee.fr/~molaeia/COURS/TP3java/index>

Traces d'exécution:

(Taux de change google à gauche, taux de change de notre convertisseur à droite)

The figure consists of four vertically stacked screenshots. Each screenshot shows a Google search result for a currency conversion on the left and a window of the 'Convertisseur EUR/USD ...' application on the right. In all cases, the input is 69 Euros.

- Top screenshot:** Conversion to US Dollars (USD). Google shows 82.36 Dollars américains. The application shows 82.8 and a rate of 1 euro = 1.2 USD.
- Second screenshot:** Conversion to British Pounds (GBP). Google shows 59.34 Livre sterling. The application shows 82.8 and a rate of 1 euro = 0.8695274118516586 GBP.
- Third screenshot:** Conversion to Japanese Yen (JPY). Google shows 9 124,97 Yen. The application shows 9188.97 and a rate of 1 euro = 133.17352510320947 JPY.
- Bottom screenshot:** Conversion to Moroccan Dirhams (MAD). Google shows 733,21 Dirham marocain. The application shows 778.47 and a rate of 1 euro = 11.282224854741354 MAD.

Les différences minimales dans le taux de change peuvent être expliquées par la source du taux de conversion: nous utilisons [Change Magenta](#), et Google utilise Morningstar.

Quelques remarques:

Dans cette version du convertisseur, ou le but de l'exercice est de récupérer les taux de change depuis internet avec jsoup, on est parti du framework de l'exercice précédent, en y apportant quelques modifications:

Tout d'abord, l'utilisation d'arraylist. Étant un type de collection, il permet de se comporter comme un "tableau dynamique", nous permettant de ne pas avoir à gérer sa taille maximale. Il peut être rempli comme une pile, et consulté index par index comme le tableau

Ensuite: après avoir scrapé la page web de changemagenta avec jsoup, nous nous sommes rendu compte que les monnaies sont référencées en valeurs minimales et en valeurs maximales. Ainsi, il a fallu faire la moyenne de chaque valeur et celle qui suit. De plus, le taux de change étant dans l'autre sens, il a fallu passer à l'inverse.

Sinon, le reste du projet reste similaire à celui du TP précédent.

Javadoc & Références:

La javadoc ainsi que le code du TP est disponible également sur [le dépôt GitHub](#)