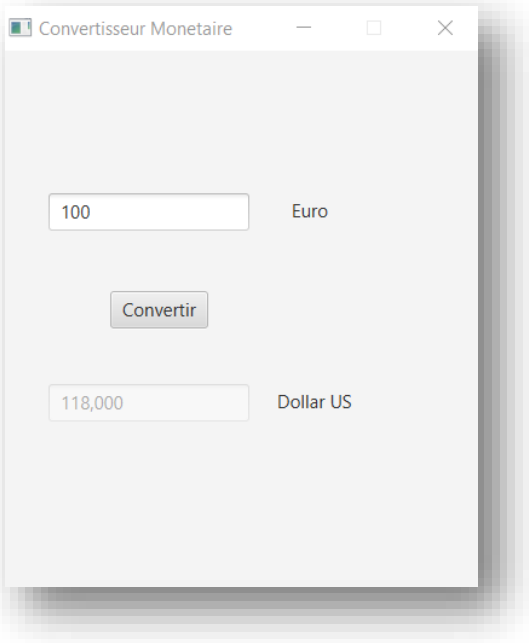
CONVERTISSEUR MONETAIRE

Java fx

1 - CONVERSION EURO - DOLLARD

Créer une application pour convertir une valeur monétaire en euros en dollards US. Le taux de conversion au 14/01/2025 est de 1,02.



Le calcul est déclenché par un click souris sur le Bouton

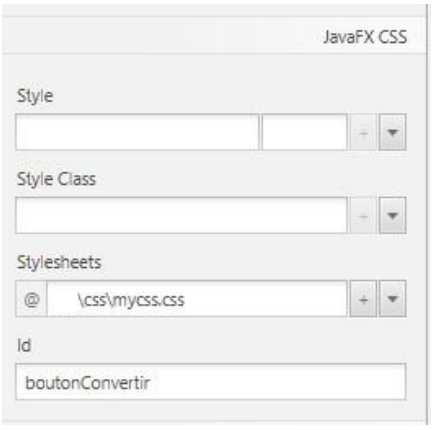
2 – STYLAGE DU BOUTON

L'image associée au composant *Button* est une image *png 40 x 40* (*convertir.png*) à stocker dans un dossier *src/images*

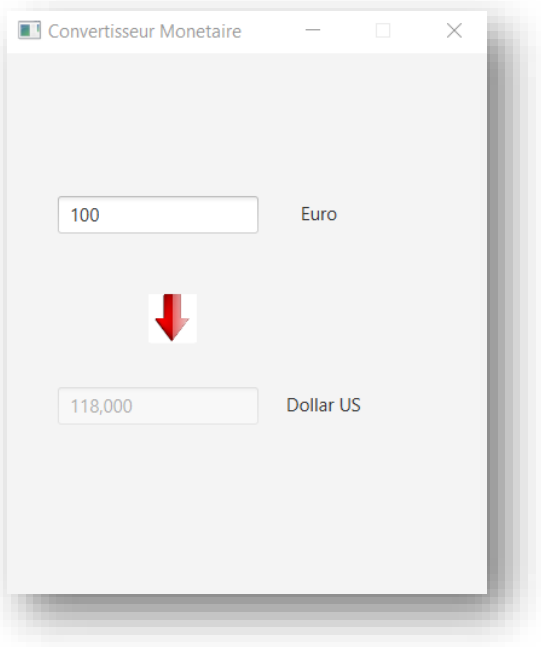
Le bouton est stylé par un fichier *css* externe.

- Créer un dossier *css* dans le dossier src
- Créer dans ce dossier *css* un fichier de type *css (cascading style sheet).*
- Le contenu de ce fichier définit les attributs de style du bouton.
- Contenu du fichier :

```
#bouttonConvertir{
-fx-background-image:
url(../images/convertir.png);
-fx-background-size: stretch;
}
```



- Associer au bouton un nom de Id de style (*bouttonConvertir*) et le chemin au fichier css.

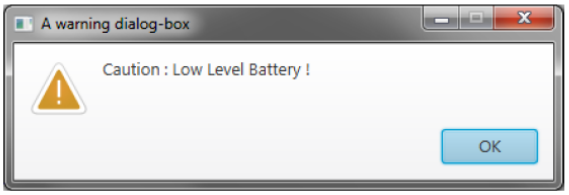


3 – AMELIORATION DE LA SECURITE

En cas de saisie d'une valeur non numérique, la conversion *String/Double* ne peut se faire et déclenche une exception de type *NumberFormatException*.

Modifier l'application en interceptant cette exception dans un bloc *try/catch* qui provoque l'affichage d'une fenêtre d'alerte.

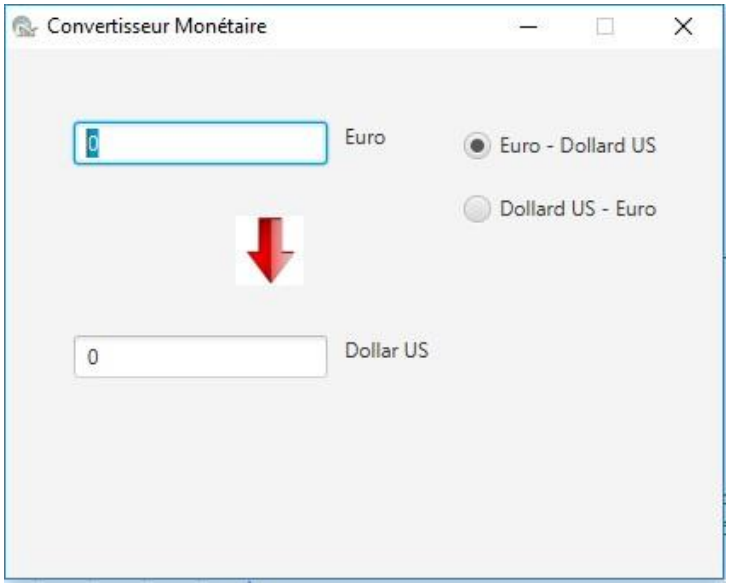
```
Alert dialogW = new Alert(AlertType.WARNING);

dialogW.setTitle("A warning dialog-box");
dialogW.setHeaderText(null);  // No header
dialogW.setContentText("Caution : Low Level Battery !");
dialogW.showAndWait();
```



4 – CONVERSION BIDIRECTIONNELLE

Rajouter sur l'IHM deux *RadioButton* pour choisir le sens de conversion.

L'exclusivité entre les deux *RadioButton* est assurée en saisissant le même nom de *Toggle Group* dans leurs propriétés.



- Mettre en place un listener sur l'évènement *Action*
- Dans la méthode associée :
- Modifier le taux de conversion et l'affichage des labels

Remarque :

L'état d'un *RadioButton* est lié à la propriété *checked*

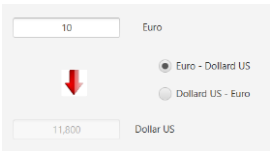
Exemple :

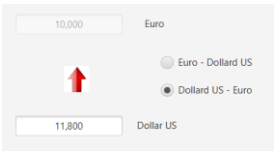
If(radioButtonEuroDollard.isSelected() == true){

…

Vous pouvez aussi réaliser une transition de type ***RotateTransition*** qui permettra de changer l'orientation de la flèche pour indiquer le sens de la convention.

```
  public RotateTransition rotation;
…
Dans initialize() :
    rotation = new RotateTransition(Duration.seconds(0.5), buttonConvertion);
…
Au moment d'appliquer la transition
    rotation.setByAngle(180);
    rotation.play();
```

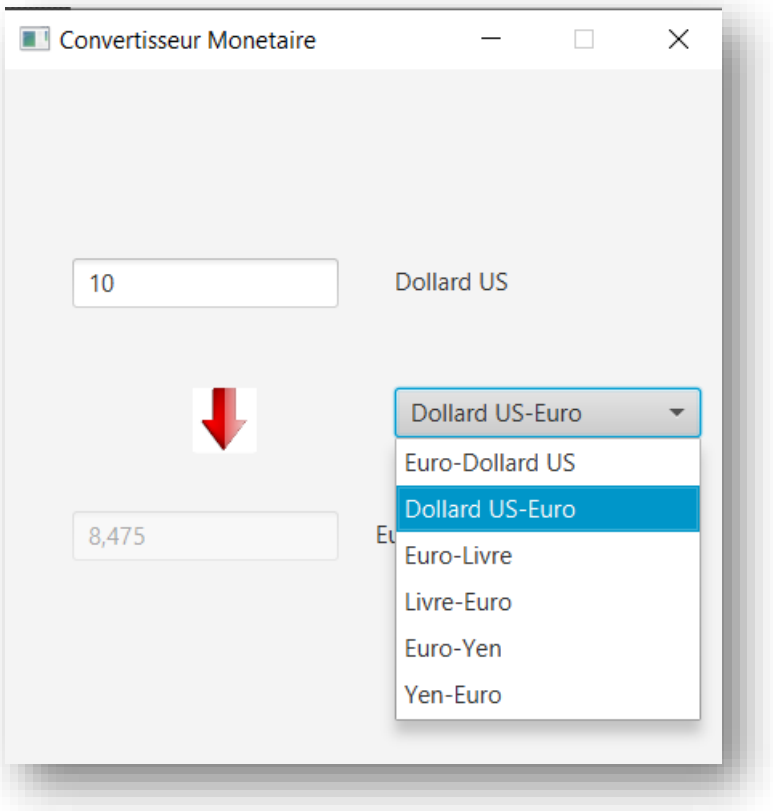




5 - GENERALISATION

Généraliser en rajoutant d'autres devises.

Dans cette généralisation, les *radioButton* peuvent être remplacés par un *ComboBox*. L'évènement à écouter par le *listener* est *Action*



Quelques règles d'utilisation du *ComboBox*.

Lorsque le *ComboBox* est créé avec le concepteur Graphique de *SceneBuilder*, On ne peut pas définir les données qui lui sont liées dans le concepteur graphique, mais dans le code du *controleur*.

Ces données sont regroupées sous la forme d'une collection *ObservableList* de *String*.

Remplissage des données:

```
comboSelection.getItems().add("une donnée");
```

- *comboSelection.getItems()* récupère la référence de la liste associée au *comboBox comboSelection*.
- La méthode *add* ajoute une *String* à la liste.

Initialisation du combobox:

```
comboSelection.setValue(comboSelection.getItems().get(0));
```

Initialisation avec la méthode *setValue* à la première valeur de la liste.

Récupération d'un valeur sélectionnée

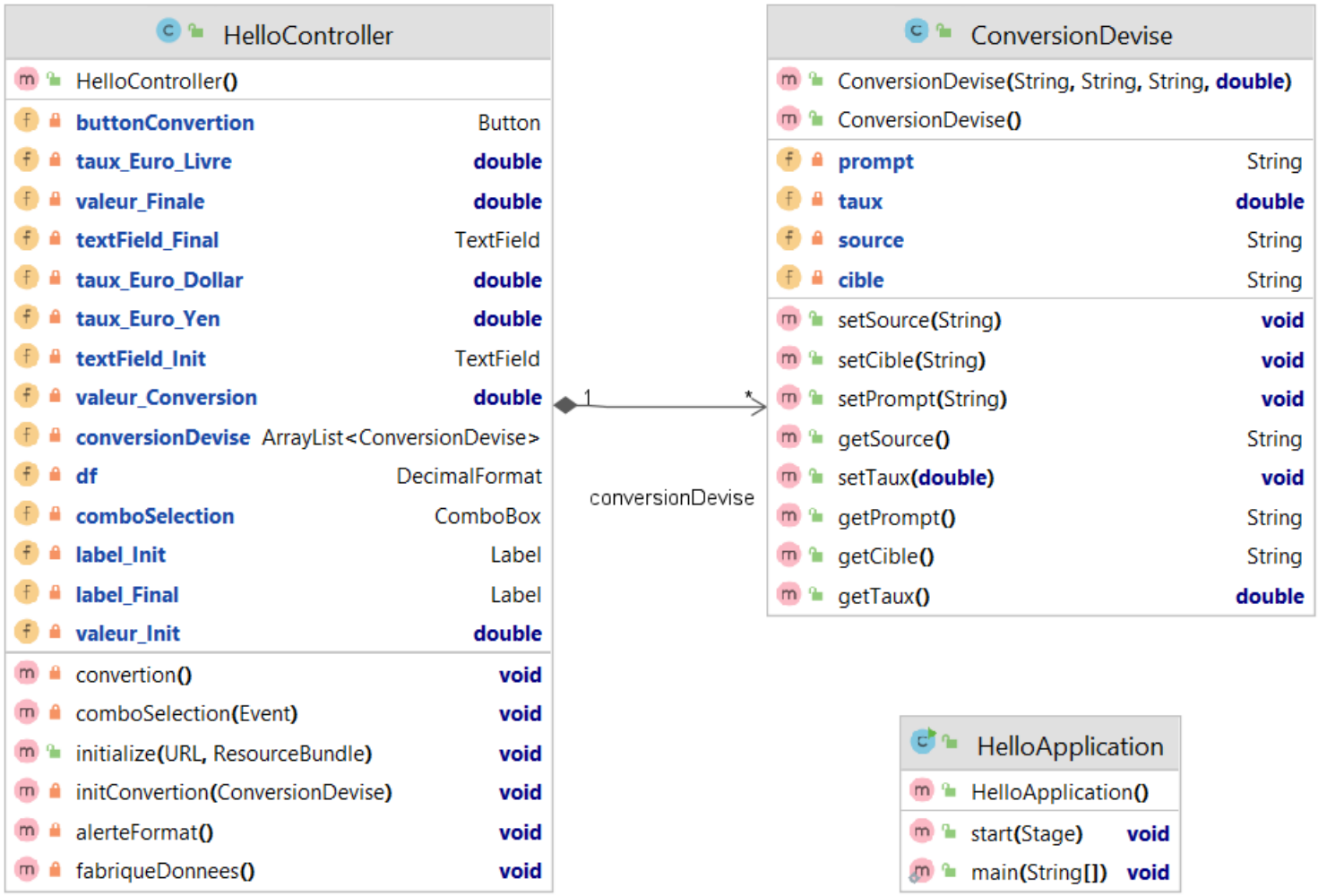
```
String ValeurSelectionnee = comboSelection.getValue()
```

6 – APPROCHE OBJET

Le taux de conversion peut être englobé dans un objet, l'ensemble des données dans un tableau d'objets.

Proposition de classe modèle

```
public class ConversionDevise {

    private String prompt;
    private String source;
    private String cible;
    private double Taux;

....
```


HelloController
HelloController()
buttonConvertion Button
taux_Euro_Livre double
valeur_Finale double
textField_Final TextField
taux_Euro_Dollar double
taux_Euro_Yen double
textField_Init TextField
valeur_Conversion double
conversionDevise ArrayList<ConversionDevise>
df DecimalFormat
comboSelection ComboBox
label_Init Label
label_Final Label
valeur_Init double
convertion() void
comboSelection(Event) void
initialize(URL, ResourceBundle) void
initConvertion(ConversionDevise) void
alerteFormat() void
fabriqueDonnees() void
1
*
conversionDevise
ConversionDevise
ConversionDevise(String, String, String, double)
ConversionDevise()
prompt String
taux double
source String
cible String
setSource(String) void
setCible(String) void
setPrompt(String) void
getSource() String
setTaux(double) void
getPrompt() String
getCible() String
getTaux() double
HelloApplication
HelloApplication()
start(Stage) void
main(String[]) void