

ANDROID – TP1

Conversion Euro-Dollar

Créer un projet

Créer un projet Android (fr.efrei.android.eurodollar) ;

nom de l'application : Conversion EuroDollar

Activité : ConversionMonnaie

XML et l'IHM

Le SDK permet de créer des strings, des interfaces, etc. soit directement avec XML ou via un éditeur ; on peut passer de l'un à l'autre en tapant sur l'onglet en bas de page.

Choisissez « res/layout/main.xml » et vérifiez cela !

Le format XML du manifest est défini ici :

<http://developer.android.com/guide/topics/manifest/manifest-intro.html>

Création d'attributs

Android permet de créer des attributs pour les ressources (chaînes, couleurs, etc..). Ces attributs peuvent être utilisés dans la définition de l'IHM via XML ou dans le code java.

Sélectionnez le fichier « res/values/string.xml » et cliquez sur Add. Sélectionnez « Color » et entrez « myColor » comme « name » et « #3399CC » comme valeur. Créer les attributs chaînes suivants (name, value) :

(« euro », « en Euro »)

(« dollar », « en dollar »)

Vérifier que tout a été bien ajouté dans le fichier XML. Vous pouvez même ajouter un 3^{ème} élément à la main : (« *calc* », « *calculer* »)

IHM

Sélectionnez « *res/layout/main.xml* » et ouvrez l'éditeur graphique Android en double-cliquant. L'éditeur vous permet de créer un IHM en utilisant le « Drag and Drop ». Vous pourriez vérifier que ce qui est ajouté niveau visuel se retrouve bien décrit dans le fichier XML correspondant (voir les onglets en bas de page).

1. Cliquer (bouton droit) sur « Hello World, Hello » et supprimer le.
2. Ajouter une zone pour rentrer du texte (la valeur en € ou en \$)
3. Ajouter 2 Radio boutons appartenant au même « Radio Group »
 - i. Sélectionner tout d'abord le « Radio Group » de « Form Widgets »
 - ii. Ajouter deux boutons radio simples à ce groupe (Euro, Dollar)
4. Ajouter un bouton à la fin
5. Vérifier le fichier XML (lisez-le attentivement)

Editer les propriétés de l'IHM

On peut changer la plupart des propriétés en utilisant le bouton droit sur un widget. On peut aussi les éditer via le fichier XML (plus rapide à réaliser en fait).

1. Ouvrez le fichier XML (*main.xml*)
2. Supprimer la ligne « *android:text= EditText* » et vérifier que cela a bel et bien été supprimé de l'IHM
3. Nommer le 1^{er} Radio bouton « Euro » et le deuxième « Dollar » en utilisant les propriétés (bouton droit sur widget et Edit Text)
4. Sélectionner la propriété « *Checked* » pour le premier bouton
5. Nommer le bouton « Calculer » et affecter « *myClickHandler* » à la propriété « *onClick* » (gestion d'évènement).
6. Mettre la propriété pour le « *input text* » à « *numberSigned* » et « *numberDecimal* »
7. Le layout choisi est le « *LinearLayout* » ; on voudrait lui affecter une couleur de fond !
8. Vérifier le tout dans le fichier « *main.xml* » (vous pouvez vérifier le sens de certaines propriétés et leurs valeurs sur le site d'Android – La référence absolue !)

Gestion d'Evènements

Pendant la génération du projet, vous avez dû définir l'activité (ConvertirMonnaie) ; le système a créé la classe Java « ConvertirMonnaie » qui hérite de « Activity » ! Il va falloir modifier le code de la façon suivante :

1. Récupérer le contenu texte avec un code « similaire » à ce qui suit
 - a. `Text = (EditText) findViewById(R.id.editText1)`
 - b. Regardez les spécifications des classes sur le site officiel ! Il faut toujours avoir ce réflexe
 - c. Que fait « `findViewById` » ?
2. Ecrire le code de gestion de l'évènement « click sur bouton »

```
public void myClickHandler(View view) {
    switch (view.getId()) {
        case R.id.button1:
            RadioButton euroButton = (RadioButton) findViewById(R.id.radio0);
            RadioButton dollarButton = (RadioButton) findViewById(R.id.radio1);
            if (text.getText().length() == 0) {
                Toast.makeText(this, "Please enter a valid number",
                    Toast.LENGTH_LONG).show();
                return;
            }

            float inputValue = Float.parseFloat(text.getText().toString());
            if (EuroButton.isChecked()) {
                text.setText(String
                    .valueOf(convertDollarToEuro(inputValue)));
                euroButton.setChecked(false);
                dollarButton.setChecked(true);
            } else {
                text.setText(String
                    .valueOf(convertEuroToDollar(inputValue)));
                dollarButton.setChecked(false);
                euroButton.setChecked(true);
            }
            break;
    }
}

// Convertir Dollar à Euro
private float convertDollarToEuro(float dollar) {
    return ...; // formule à utiliser
}
```

```
// Convertir Euro à Dollar
private float convertEuroToDollar(float euro) {
    return ...; // formule à utiliser
}
}
```

Maintenant, vous pouvez lancer et tester votre application !

1. Ouvrez les perspectives suivantes logCat , DDMS, et « File Explorer »
2. Dans « file Explorer », aller dans Data, puis app et trouver le fichier « apk » qui correspond à votre application !

Ajouter votre application à un vrai Terminal

1. Activer « USB Debugging » dans les paramètres du terminal (paramètres Application > Développement)
2. Bien sûr la version de développement devrait être celle du terminal !
3. Il faudra installer le pilote du terminal ; pour les détails techniques, lire la page suivante <http://developer.android.com/guide/developing/device.html>