

Examen Programmation Swift

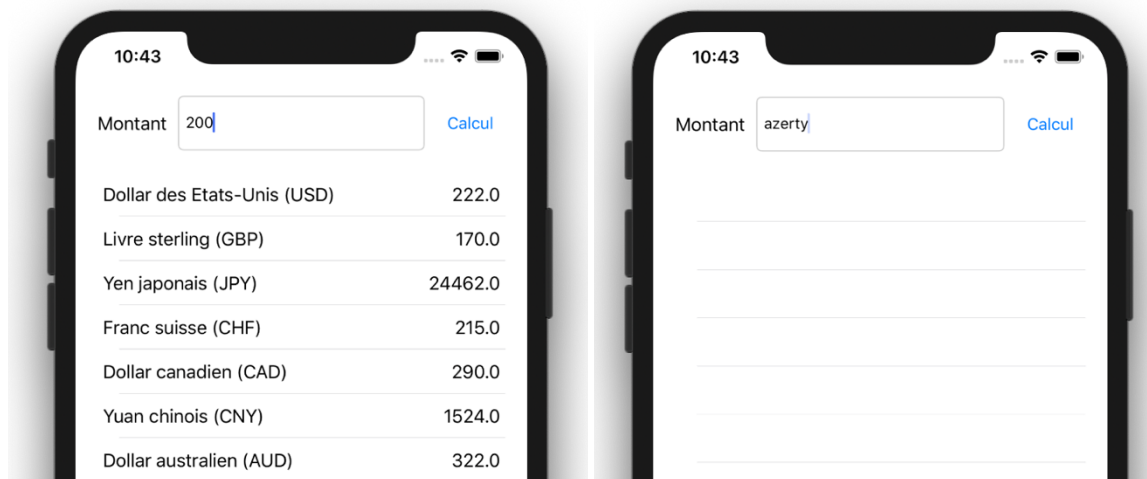
Durée 1h20 – Documents autorisés

A la fin de l'examen, une archive zip du code doit être envoyée à dginhac@u-bourgogne.fr

Exercice : Convertisseur de devises

L'objectif de cet exercice est de proposer un convertisseur de devises. Elle permet de saisir dans un champ texte une valeur. Si cette valeur est un nombre (correspondant à un montant en euros), alors l'application calcule et affiche dans un TableView la liste des conversions de ce montant dans différentes devises (voir copie d'écran à gauche) en appliquant un taux de change. Si la valeur saisie n'est pas un nombre, rien n'est affiché dans le TableView (voir copie d'écran à droite).

Par exemple, le taux de change du dollar américain (USD) est de 1.1108, ce qui donne 222 \$ pour 200 euros ($222 = 200 * 1.1108$).



Les intitulés des devises et les taux de change pratiqués seront stockés dans des variables de type tableau (voir ci-dessous). Pour vous éviter d'avoir à saisir ces 2 variables, vous pouvez télécharger un fichier swift à l'adresse suivante : <http://ginhac.com/data.html>

```
var monnaies = ["Dollar des Etats-Unis (USD)", "Livre sterling (GBP)", "Yen japonais (JPY)", "Franc suisse (CHF)", "Dollar canadien (CAD)", "Yuan chinois (CNY)", "Dollar australien (AUD)", "Real brésilien (BRL)", "Won coréen (KRW)", "Peso mexicain (MXN)", "Ringgit malaisien (MYR)", "Couronne norvégienne (NOK)", "Rouble russe (RUB)", "Couronne suédoise (SEK)", "Couronne danoise (DKK)"]

var tauxDeChange = [1.1108, 0.85105, 122.31, 1.0736, 1.4498, 7.6186, 1.6122, 4.639, 1288.37, 20.8338, 4.5041, 9.889, 68.2495, 10.545, 7.4729]
```

1 - Création du StoryBoard de l'application

Créer une nouvelle application iOS de type « Single View App ». Nommer cette application avec votre NomPrénom (Ex : GinhacDominique).

L'interface comprend le champ texte de type UITextField (la valeur de ce champ texte est accessible via la propriété « text »), le bouton Calcul et le TableView. Pour le TableView, créer la cellule prototype et inclure deux Label permettant d'afficher le nom de la devise à gauche et le montant converti à droite selon le format présenté sur les copies d'écran.

Attention de bien rajouter toutes les contraintes nécessaires pour que l'affichage soit conforme à la copie d'écran ci-dessus, quelle que soit la taille de l'iPhone.

Merci de faire valider le storyboard par l'enseignant !

2- Gestion du bouton Calcul

Dans cette question, on s'intéresse à calculer la liste des devises converties qui seront ensuite affichées dans le TableView.

Pour cela, créer une Action correspondant au clic sur le bouton Calcul. Si la valeur inscrite dans le TextField est bien un nombre réel, alors la liste des montants des devises est calculée et stockée dans un tableau de réels (ex : var montants : [Double])

Merci de faire valider le fonctionnement par l'enseignant en effectuant un affichage du tableau des montants de devises grâce à l'instruction print.

3- Remplissage du TableView

Dans cette question, on s'intéresse à créer les différentes cellules du TableView avec des données et ainsi afficher les intitulés des devises et les montants.

Pour cela, créer la classe de type TableViewCell associée au prototype de la cellule et les variables Outlet correspondant aux deux labels. Dans le ViewController principal, ajouter le protocole adéquat et écrire les 2 méthodes nécessaires pour remplir le TableView à partir du tableau des montants des devises.

Merci de faire valider le fonctionnement par l'enseignant !

4- Actualisation du TableView

Dans cette question, on s'intéresse à actualiser les valeurs affichées dans le TableView en fonction de la valeur du TextField. Modifier l'Action liée au clic sur le bouton calcul pour que le TableView soit automatiquement réactualisé. Si la valeur du TextField est un montant réel, alors la liste des devises est affichée, sinon toutes les valeurs du TableView sont effacées.

Merci de faire valider le fonctionnement par l'enseignant !

5- Question complémentaire

Pour éviter les erreurs de saisie, ajouter un Stepper permettant d'augmenter ou baisser le montant affiché dans le champ texte. On supposera que le montant à convertir est dans l'intervalle [0-5000] et que le pas d'incrément/décrémentation est de 100.

A chaque nouveau click sur le Stepper, les valeurs du TableView sont actualisées. Pas besoin de cliquer sur le bouton calcul.

Merci de faire valider le fonctionnement par l'enseignant !

