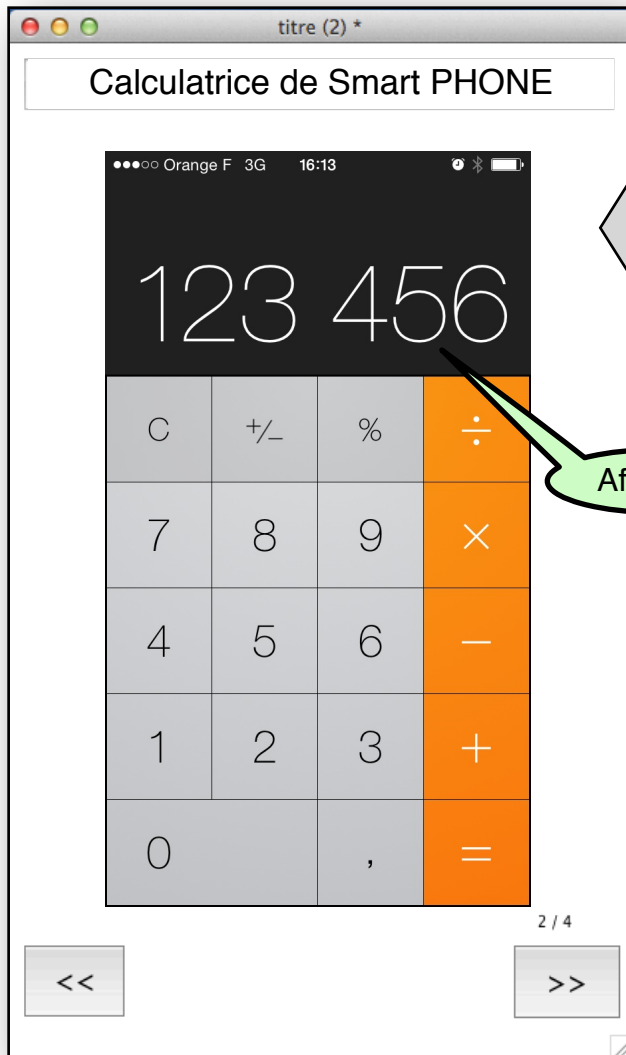


Programmer les fonctions d'une calculatrice



Exercice d'application :

Bâtir l'interface de programmation ci-contre en complétant la pile de votre session .



1 Récupérer l'image de fond sur le réseau. après création d'une zone "image aréa" et donner le chemin de l'image.

2 Tracer des boutons transparents au dessus de l'image sur les boutons "C" puis les Boutons "9" à "0" et ",". Attention au "comportement" et "0" et "," qui ne sont pas à afficher systématiquement.

3 Créer un champ fond noir avec texte blanc sur la zone "Afficheur" et verrouiller le texte dans les propriétés du champ.

4 Compléter les scripts de chaque bouton de "0" à "9" afin de remonter les valeurs dans le champ "Afficheur" et effacer la zone avec "C". ne pas confondre Put into, before, after....

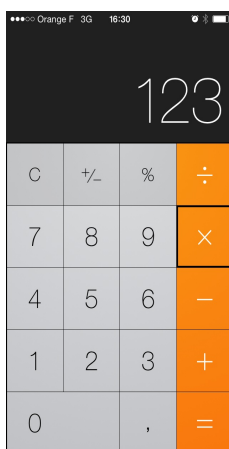
5 Tracer des boutons transparents au dessus des boutons "+/-" puis le boutons "%", compléter les scripts de ces deux boutons.

6 Tracer des boutons au dessus des 4 opérations et "=". Paramétrer les contours pour qu'ils puissent être vus épais lorsqu'ils seront actionnés lors des calculs. (voir les propriétés)

Show name, Opaque, border, border width, border color.

voir **set the showBorder of button "n" to false (ou true) vrai ou faux** (modifier par script les propriétés de bouton)

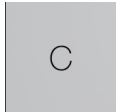
7 Débuter le codage des boutons des 4 opérateurs et le bouton "Egal" (après quelques essais les scripts "remonteront" dans un script de carte)



Voir détails pages suivante

Premières actions de base à programmer

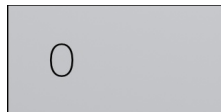
Afficheur



C mettre "0" **dans** AFFICHEUR
mettre les variables à Zéro (varv



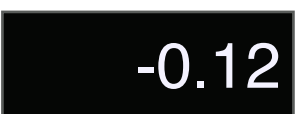
, mettre "." **après** si pas de virgule (varv =0)
mettre 1 **dans** varv (varv = 1)
mettre rien si varv =1



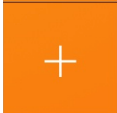
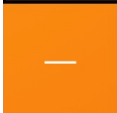
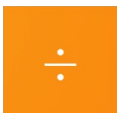
0 Mettre "0" **après** si Afficheur différent de "0"
Mettre rien si afficheur = 0
Mettre 0 **après** si Afficheur = 0 et varv =1



+/- Multiplier Afficheur par -1



% Diviser Afficheur par 100



+ - x Paramétrer les contours pour qu'ils puissent être vus épais lorsqu'ils seront actionnés lors des calculs
`set the showBorder of button "x" to true`
mettre les 3 autres à "false"
sur chaque bouton opérateur.

= en actionnant =
`set the showBorder of button "x" to false`
mettre aussi les 3 autres à "false"
sur chaque bouton opérateur.

7	8	9
4	5	6
1	2	3

si affichage = "0" mettre 1 ou 2 ou 3
dans affichage.

si affichage différent de "0" mettre 1 ou 2 ou 3 **après** affichage.

