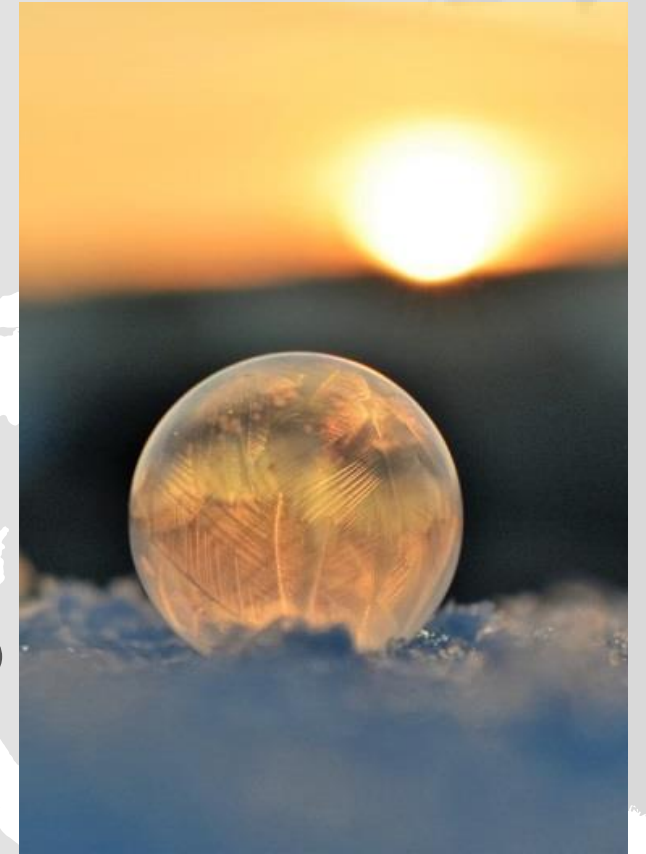


# App Inventor

liste de listes



Correction  
de l'exercice  
sous-listes de mots ?





# TUTO SUR LES LISTES ET LES BOUCLES

- Ce tutoriel vidéo fait partie d'un cours <https://onvaessayer.org> dans les notions de base sur App Inventor
    - <https://onvaessayer.org/appinventor?app=baseApps>
  - Cette vidéo fait partie de la série suivante
    - Intro : <https://onvaessayer.org/appinventor?video=disMoi>
    - V1 : <https://onvaessayer.org/appinventor?video=disMoiV1>
    - V2 : <https://onvaessayer.org/appinventor?video=disMoiV2>
    - V2exo : <https://onvaessayer.org/appinventor?video=disMoiV2exo>
    - V3 : <https://onvaessayer.org/appinventor?video=disMoiV3>
    - V4 : <https://onvaessayer.org/appinventor?video=disMoiV4>
- et fait référence à une introduction aux listes
- <https://onvaessayer.org/appinventor?video=listesN1>



# V2a : EXTENSION

- Rechercher plusieurs mots pour identifier la réponse
- Par exemple **le** et **jeune** pour trouver Lucas :
  - **le** plus **jeune** va répondre "Lucas"
  - **la** plus **jeune** va répondre "je ne sais pas"
- Par exemple avec le tableau suivant
  - à gauche la liste des réponses
  - à droite la liste des mots clefs qui est une liste de listes car chaque élément est lui même une liste

réponses	Liste de listes de mots clefs
Clara	la , belle
Lucas	le , jeune
Emmanuel	plus , rapide
Lisa	dynamique
Pierre	vieux

# V2a : EXTENSION



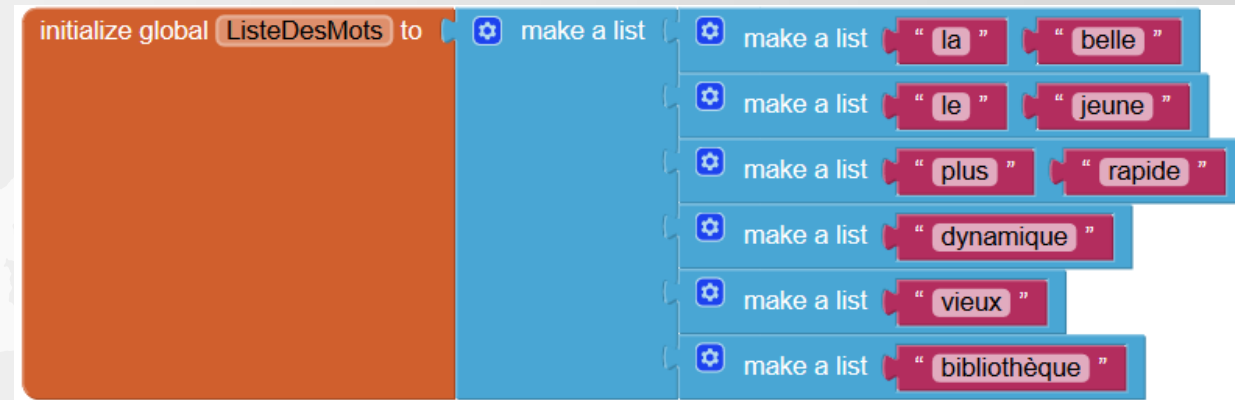
- Plusieurs mots doivent être présents pour une personne
- Par exemple **le** et **jeune** trouver Lucas :
  - **le** plus **jeune** va répondre "Lucas"
  - **la** plus **jeune** va répondre "je ne sais pas«

Avant de faire cette extension, enregistrez la version de votre application avec un nom dont vous vous rappellerez nous la réutiliserons ensuite.

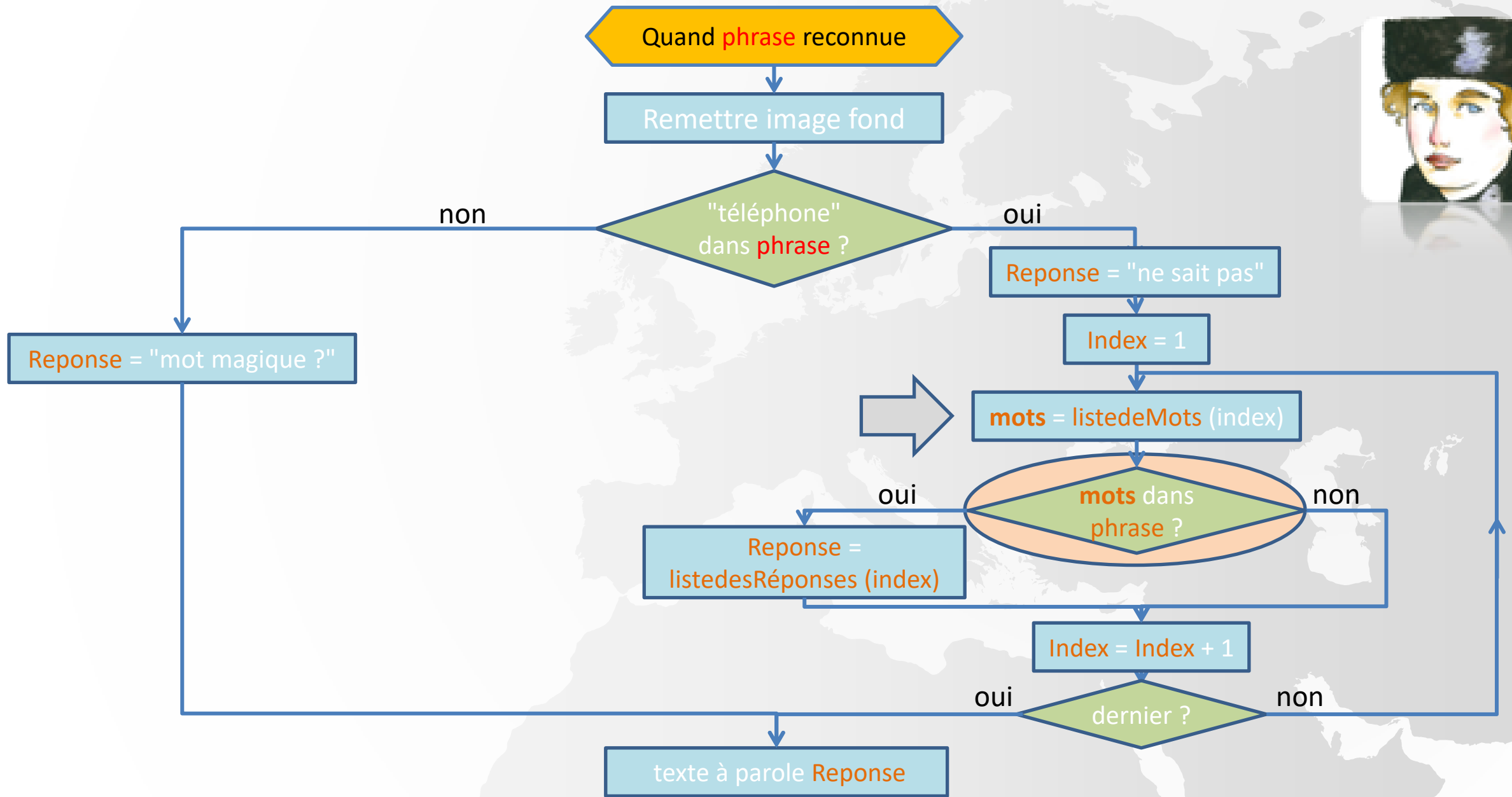


# V2a : EXTENSION

- la **liste de mots** à reconnaître devient une **liste de listes de mots**,



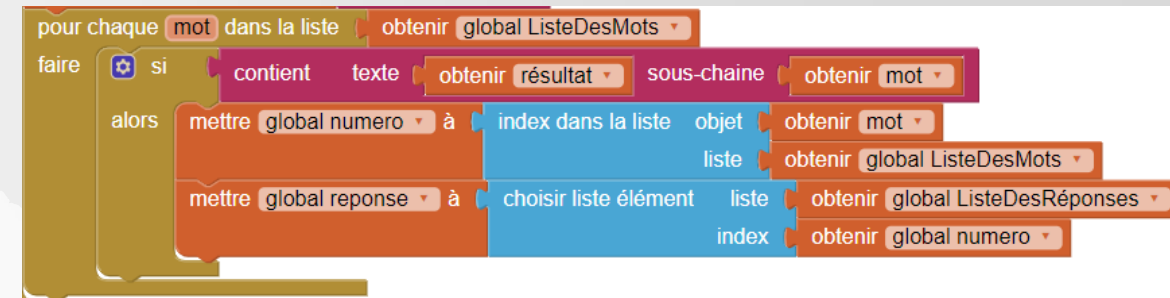
- la liste des réponses ne change pas.



# V2 : PSEUDO-CODE



- Quand phrase reconnue
  - afficher dans label et remettre image initiale
  - Si phrase contient "téléphone"
    - reponse = "je ne sais pas"
    - Pour chaque mot dans listeDesMots
      - Si phrase contient mot :
        - numero = index de mot dans listeDesMots
        - reponse = choisir dans listeDesRéponses l'élément à l'index numero
  - Sinon
    - reponse = "tu ne m'as pas dit le mot magique"
- Vocaliser reponse



# V2A : PSEUDO-CODE



- Quand phrase reconnue
  - afficher dans label et remettre image initiale
  - Si phrase contient "téléphone"
    - reponse = "je ne sais pas"
    - Pour chaque sousListeMots dans listeDesMots
      - Si phrase contient sousListeMots :
        - numero = index de sousListeMots dans listeDesMots
        - reponse = choisir dans listeDesRéponses l'élément à l'index numero
  - Sinon
    - reponse = "tu ne m'as pas dit le mot magique"
  - Vocaliser reponse



# V2A : PSEUDO-CODE



- Quand phrase reconnue
  - afficher dans label et remettre image initiale
  - Si phrase contient "téléphone"
    - reponse = "je ne sais pas"
    - Pour chaque sousListeMots dans listeDesMots
      - Si phrase contient sousListeMots :  ???
      - numero = index de sousListeMots dans listeDesMots
      - reponse = choisir dans listeDesRéponses l'élément à l'index numero
  - Sinon
    - reponse = "tu ne m'as pas dit le mot magique"
- Vocaliser reponse

# V2A : PSEUDO-CODE



- Quand phrase reconnue
  - afficher dans label et remettre image initiale
  - Si phrase contient "téléphone"
    - reponse = "je ne sais pas"
    - Pour chaque sousListeMots dans listeDesMots
      - phraseContientSousListeMots = vrai si contient faux sinon
      - Si phraseContientSousListeMots :
        - numero = index de sousListeMots dans listeDesMots
        - reponse = choisir dans listeDesRéponses l'élément à l'index numero
  - Sinon
    - reponse = "tu ne m'as pas dit le mot magique"
- Vocaliser reponse

# V2A : PSEUDO-CODE



- Calcul de la variable : **phraseContientSousListeMots**
  - **vrai** si la phrase contient chacun des mots de la liste (tous)
  - **faux** sinon
- On initialise cette variable à **vrai**  
et si un seul des mots n'est PAS trouvé, on la met à **faux**

```
phraseContientSousListeMots = vrai
Pour chaque mot dans sousListeDesMots
| si phrase ne contient pas mot
| | phraseContientSousListeMots = faux
```

# V2A : PSEUDO-CODE



- Quand phrase reconnue
  - afficher dans label et remettre image initiale
  - Si phrase contient "téléphone"
    - reponse = "je ne sais pas"
    - Pour chaque sousListeMots dans listeDesMots
      - phraseContientSousListeMots = vrai
      - Pour chaque mot dans sousListeDesMots
        - si phrase ne contient pas mot
        - phraseContientSousListeMots = faux
    - Si phraseContientSousListeMots :
      - numero = index de sousListeMots dans listeDesMots
      - reponse = choisir dans listeDesRéponses l'élément à l'index numero
  - Sinon ...

# V2A : PSEUDO-CODE



- Quand phrase reconnue  
afficher dans label et remettre image initiale  
Si phrase contient "téléphone"

reponse = "je ne sais pas"

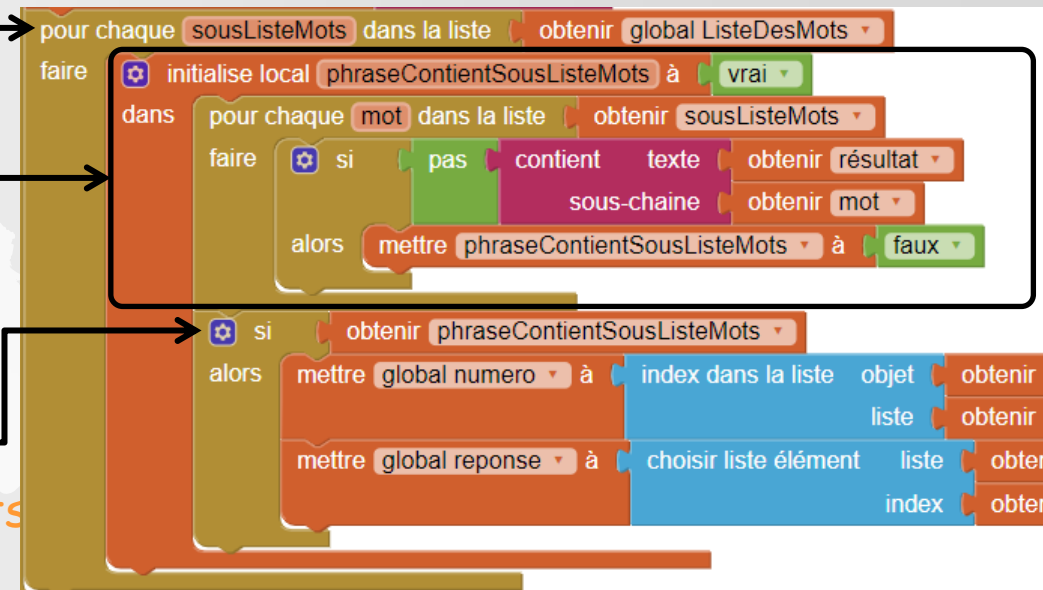
Pour chaque sousListeMots dans listeDesMots

```
phraseContientSousListeMots = vrai
Pour chaque mot dans sousListeDesMots
  si phrase ne contient pas mot
    phraseContientSousListeMots = faux
```

• Si phraseContientSousListeMots :

- numero = index de sousListeMots dans listeDesMots
- reponse = choisir dans listeDesRéponses l'élément à l'index numero

• Sinon ...



# V2A : UTILISATION D'UNE PROCÉDURE



1° solution : utilisation d'une procédure qui calcule la variable `phraseContientSousListeMots`

- Quand phrase reconnue

afficher dans label et remettre image initiale

Si phrase contient "téléphone"

reponse = "je ne sais pas"

Pour chaque `sousListeMots` dans `listeDesMots`

Appeler `ContientListeMots`

Si `phraseContientSousListeMots` :

numero = index de `sousListeMots` dans `listeDesMots`

reponse = choisir dans `listeDesRéponses`  
l'élément à l'index `numero`

Sinon

reponse = "tu ne m'as pas dit le mot magique"

- Vocaliser reponse

Procédure qui met à jour la variable  
`phraseContientSousListeMots`

`phraseContientSousListeMots` = vrai  
Pour chaque mot dans `sousListeDesMots`  
si phrase ne contient pas mot  
`phraseContientSousListeMots` = faux

# V2A : UTILISATION D'UNE PROCÉDURE



2° solution : utilisation d'une procédure qui renvoie la valeur à tester

- Quand phrase reconnue

- afficher dans label et remettre image initiale

- Si phrase contient "téléphone"

- reponse = "je ne sais pas"

- Pour chaque sousListeMots dans listeDesMots

- Si phraseContientSousListeMots :

- numero = index de sousListeMots dans listeDesMots

- reponse = choisir dans listeDesRéponses  
l'élément à l'index numero

- Sinon

- reponse = "tu ne m'as pas dit le mot magique"

- Vocaliser reponse

Procédure qui renvoie vrai ou faux

motsTrouvés = vrai

Pour chaque mot dans sousListeDesMots

si phrase ne contient pas mot

motsTrouvés = faux

Renvoyer motsTrouvés



# V2A : UTILISATION D'UNE PROCÉDURE



- Quand phrase reconnue

- afficher dans label et remettre image initiale

- Si phrase contient "téléphone"

- reponse = "je ne sais pas"

- Pour chaque sousListeMots dans listeDesMots

- Si phraseContientSousListeMots :

- numero = index de sousListeMots dans listeDesMots

- reponse = choisir dans listeDesRéponses  
l'élément à l'index numero

- Sinon

- reponse = "tu ne m'as pas dit le mot magique"

- Vocaliser reponse

Procédure qui renvoie vrai ou faux

motsTrouvés = vrai

Pour chaque mot dans sousListeDesMots

si phrase ne contient pas mot

motsTrouvés = faux

Renvoyer motsTrouvés



# V2A : UTILISATION D'UNE PROCÉDURE



```
initialise local phraseContientSousListeMots à vrai
dans
  pour chaque mot dans la liste obtenir sousListeMots
  faire
    si pas contient texte obtenir résultat
      sous-chaine obtenir mot
    alors mettre phraseContientSousListeMots à faux
  si obtenir phraseContientSousListeMots
  alors
    mettre global numero à index dans la liste objet obtenir sousListeMots
    liste obtenir global ListeDesMots
    mettre global reponse à choisir liste élément liste obtenir global ListeDesRéponses
    index obtenir global numero
```

- Utilisation d'une procédure qui renvoie **vrai** ou **faux** selon qu'une phrase contient ou non une liste de mots

# V2A : UTILISATION D'UNE PROCÉDURE



```
initialise local phraseContientSousListeMots à vrai
dans
  pour chaque mot dans la liste obtenir sousListeMots
  faire
    si pas contient texte obtenir résultat
      sous-chaine obtenir mot
    alors mettre phraseContientSousListeMots à faux
  si obtenir phraseContientSousListeMots
  alors
    mettre global numero à index dans la liste objet obtenir sousListeMots
    liste obtenir global ListeDesMots
    mettre global reponse à choisir liste élément liste obtenir global ListeDesRéponses
    index obtenir global numero
```

```
à phraseContientMots listeMotsATrouver phrase
résultat initialise local motsTrouves à vrai
dans
  faire
    pour chaque mot dans la liste obtenir listeMotsATrouver
    faire
      si pas contient texte obtenir phrase
        sous-chaine obtenir mot
      alors mettre motsTrouves à faux
  résultat obtenir motsTrouves
```

- Utilisation d'une procédure qui renvoie **vrai** ou **faux** selon qu'une phrase contient ou non une liste de mots

# V2A : UTILISATION D'UNE PROCÉDURE



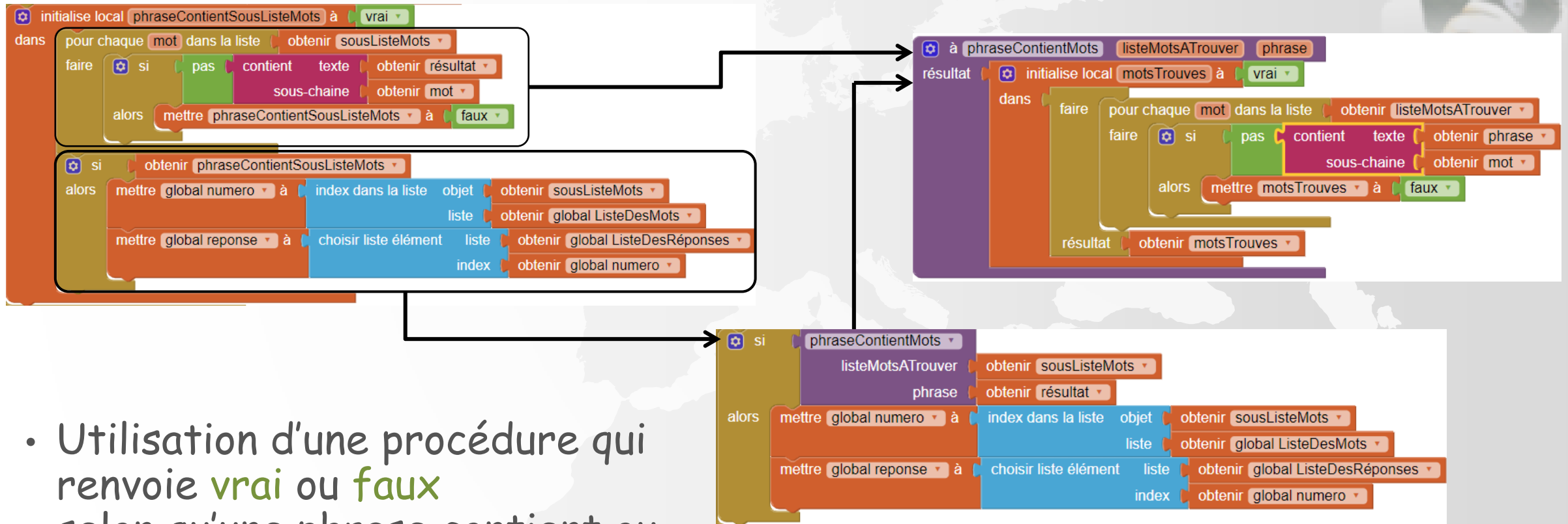
```
initialise local phraseContientSousListeMots à vrai
dans
  pour chaque mot dans la liste obtenir sousListeMots
  faire
    si pas contient texte obtenir résultat
      sous-chaine obtenir mot
    alors mettre phraseContientSousListeMots à faux
  si obtenir phraseContientSousListeMots
  alors
    mettre global numero à index dans la liste objet obtenir sousListeMots
    liste obtenir global ListeDesMots
    mettre global reponse à choisir liste élément liste obtenir global ListeDesRéponses
    index obtenir global numero
```

```
à phraseContientMots listeMotsATrouver phrase
résultat
  initialise local motsTrouves à vrai
  dans
    faire
      pour chaque mot dans la liste obtenir listeMotsATrouver
      faire
        si pas contient texte obtenir phrase
          sous-chaine obtenir mot
        alors mettre motsTrouves à faux
    résultat obtenir motsTrouves
```

```
si phraseContientMots
  listeMotsATrouver obtenir sousListeMots
  phrase obtenir résultat
  alors
    mettre global numero à index dans la liste objet obtenir sousListeMots
    liste obtenir global ListeDesMots
    mettre global reponse à choisir liste élément liste obtenir global ListeDesRéponses
    index obtenir global numero
```

- Utilisation d'une procédure qui renvoie **vrai** ou **faux** selon qu'une phrase contient ou non une liste de mots

# V2A : UTILISATION D'UNE PROCÉDURE



- Utilisation d'une procédure qui renvoie **vrai** ou **faux** selon qu'une phrase contient ou non une liste de mots

# V2A : CODE



```
initialise global numero à "0"  
initialise global reponse à ""  
quand Cadre1 .Touché  
  x y touchedAnySprite  
  faire  
    mettre Screen1 . Image de fond à "BouleDeCristal2.jpg"  
    appeler Reconnaissance_vocale1 .Obtenir texte
```

```
quand Reconnaissance_vocale1 .Après obtention texte  
  résultat  
  faire  
    mettre Label1 . Texte à obtenir résultat  
    mettre Screen1 . Image de fond à "BouleDeCristal.jpg"  
    si contient texte obtenir résultat sous-chaine "éléphone"  
    alors  
      mettre global reponse à "Je ne sais pas"  
      pour chaque sousListeMots dans la liste obtenir global ListeDesMots  
      faire  
        si phraseContientMots listeMotsATrouver obtenir sousListeMots phrase obtenir résultat  
        alors  
          mettre global numero à index dans la liste objet obtenir sousListeMots liste obtenir global ListeDesMots  
          mettre global reponse à choisir liste élément liste obtenir global ListeDesRéponses index obtenir global numero  
      sinon  
        mettre global reponse à "tu ne mas pas dit le mot magique."  
    appeler Texte_à_parole1 .Parler message obtenir global reponse
```

```
initialise global ListeDesMots à  
  créer une liste  
  créer une liste "la" "belle"  
  créer une liste "le" "jeune"  
  créer une liste "plus" "rapide"  
  créer une liste "dynamique"  
  créer une liste "vieux"  
  créer une liste "bibliothèque"
```

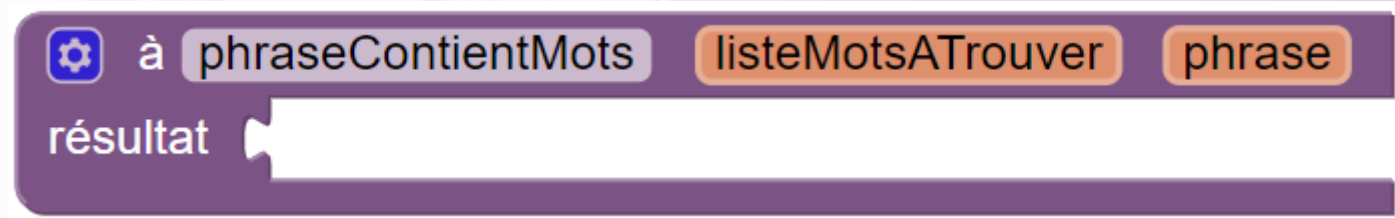
```
initialise global ListeDesRéponses à  
  créer une liste  
  Clara  
  Lucas  
  Emmanuel  
  Lisa  
  Pierre  
  Marguerite Yourcenar
```

```
à phraseContientMots listeMotsATrouver phrase  
résultat  
  initialise local motsTrouves à vrai  
  dans  
    faire  
      pour chaque mot dans la liste obtenir listeMotsATrouver  
      faire  
        si pas contient texte obtenir phrase sous-chaine obtenir mot  
        alors  
          mettre motsTrouves à faux  
      résultat  
    obtenir motsTrouves
```

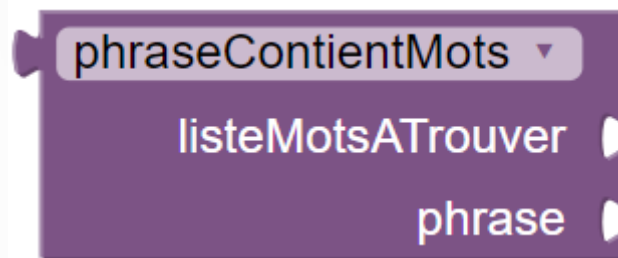
# V2A : PROCÉDURE



- Les procédures qui renvoient une valeur ont une encoche



- On les appelle avec des blocs qui retournent une valeur

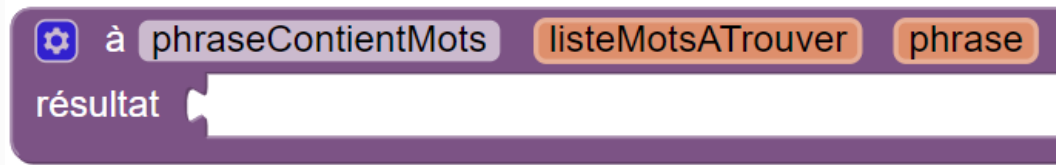


- Avec ici deux paramètres
  - la (sous) liste des mots à trouver
  - La phrase dans laquelle il faut trouver ces mots

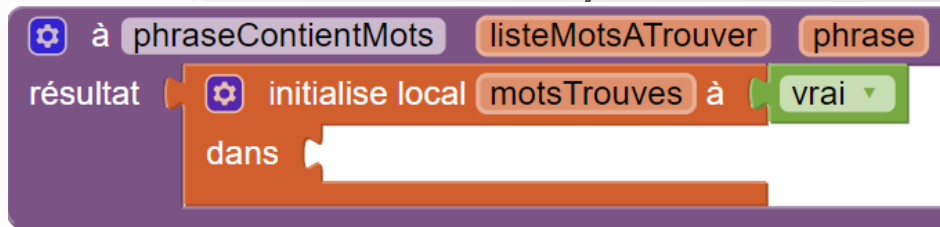
# V2A : PROCÉDURE



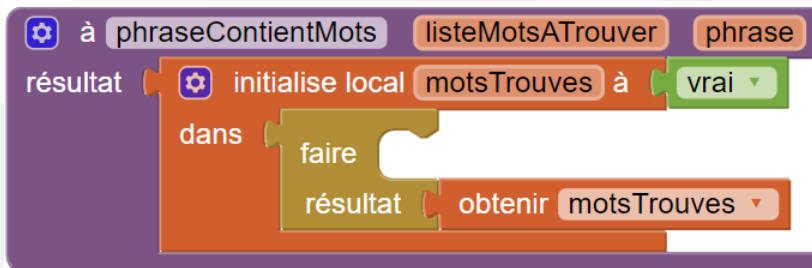
- Les procédures qui renvoient une valeur ont une encoche



- On peut déclarer des variables locales, dont celle qui sera retournée



- Puis insérer un bloc de contrôle prévu pour retourner une valeur et dans lequel on met l'algorithme

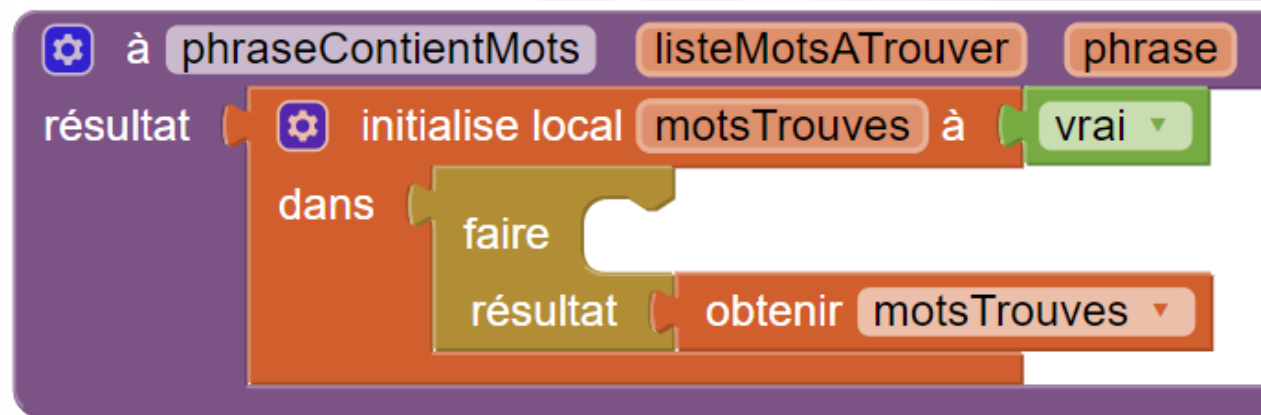




# V2A : PROCÉDURE



- Puis insérer un bloc de contrôle prévu pour retourner une valeur et dans lequel on met l'algorithme

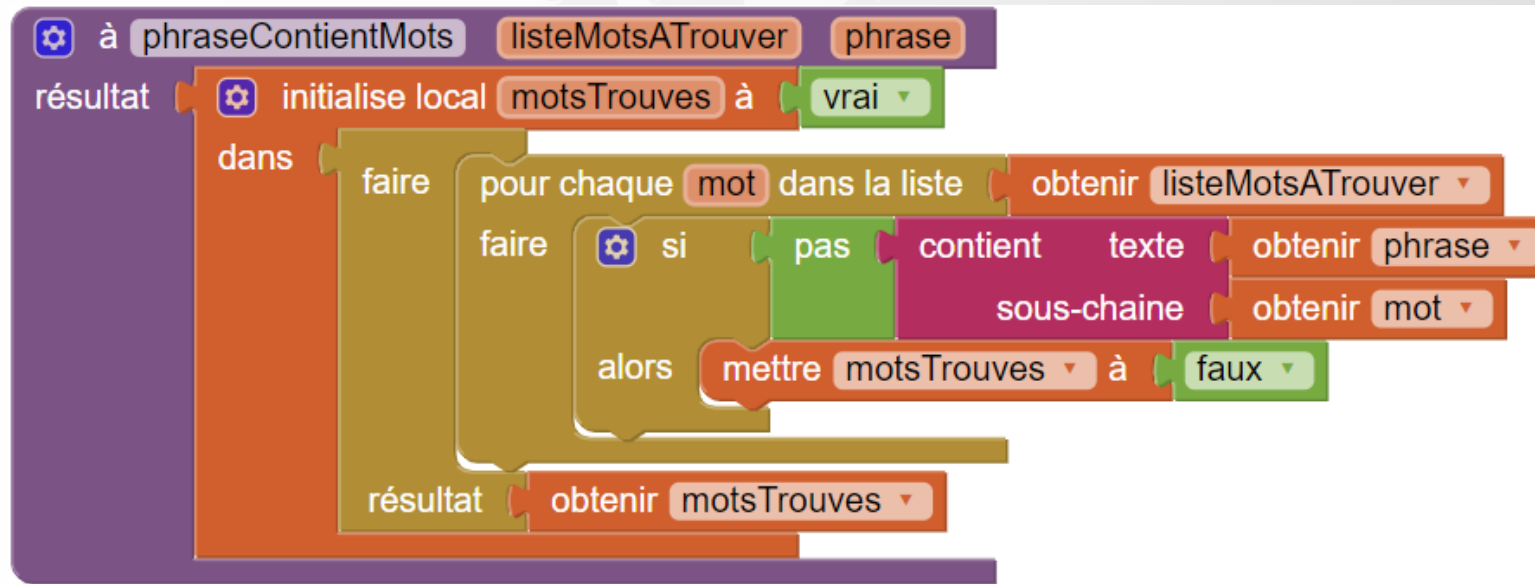




# V2A : PROCÉDURE



- Puis insérer un bloc de contrôle prévu pour retourner une valeur et dans lequel on met l'algorithme



# V2A : PROCÉDURE



- Puis insérer un bloc de contrôle prévu pour retourner une valeur et dans lequel on met l'algorithme

```
à phraseContientMots listeMotsATrouver phrase
résultat initialise local motsTrouves à vrai
dans
  faire
    si pas est une liste? objet obtenir listeMotsATrouver
    alors
      mettre listeMotsATrouver à diviser texte obtenir listeMotsATrouver
      à “ , ”
    pour chaque mot dans la liste obtenir listeMotsATrouver
    faire
      si pas contient texte majuscule obtenir phrase
      sous-chaine majuscule couper obtenir mot
      alors
        mettre motsTrouves à faux
  résultat obtenir motsTrouves
```



# V2A : RISQUES D'ERREUR

- Que se passe t-il si vous avez oublié de déclarer un des élément de la liste comme une liste ?
  - Par exemple pour les 2 derniers éléments
- ça pourrait marcher ...





# V2A : RISQUES D'ERREUR

- Que se passe t-il si vous avez oublié de déclarer un des élément de la liste comme une liste ?

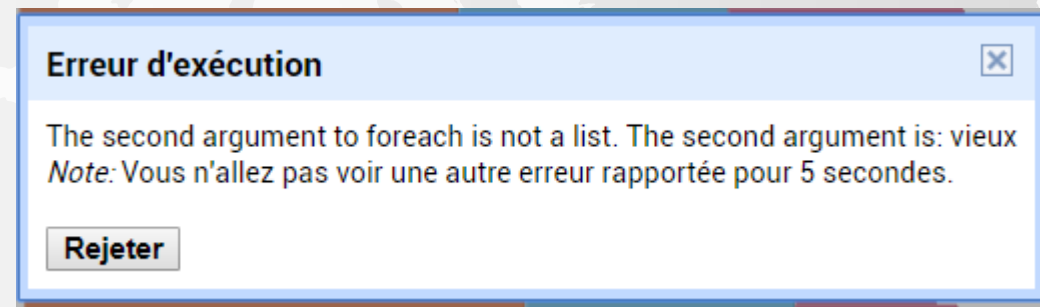
- Par exemple pour les 2 derniers éléments

- ça pourrait marcher ...



- Eh bien Non : **Erreur**

Les commandes de listes ne peuvent traiter que des variables de type liste





# V2A : RISQUES D'ERREUR

- Comment faire si on veut quand même traiter les sous-listes et les mots seuls, sans avoir à ajouter





# V2A : RISQUES D'ERREUR

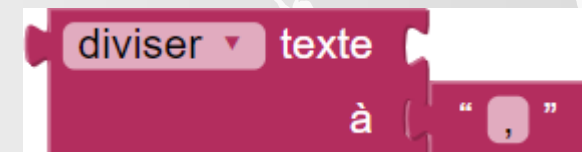
- Comment faire si on veut quand même traiter les sous-listes et les mots seuls, sans avoir à ajouter



- Il faut tester si la variable lue est une liste



- Et dans la négative convertir cette variable en liste, par exemple avec la virgule comme séparateur (commande diviser ou split en anglais)



# V2A : RISQUES D'ERREUR



- Autres risque d'erreur
  - La présence de blancs au début ou à la fin d'un mot clef (on ne les voit pas)
  
- La différence entre minuscules et majuscules

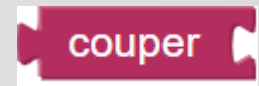
# V2A : RISQUES D'ERREUR



- Autres risque d'erreur
  - La présence de blancs au début ou à la fin d'un mot clef (on ne les voit pas)

Solution :

utiliser la command trim (en anglais) ou couper (en français)  
qui supprime les blancs au début et à la fin



- La différence entre minuscules et majuscules



# V2A : RISQUES D'ERREUR

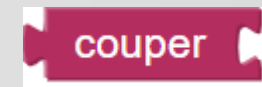


- Autres risque d'erreur

- La présence de blancs au début ou à la fin d'un mot clef (on ne les voit pas)

Solution :

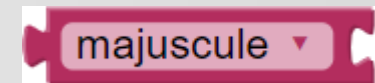
utiliser la command trim (en anglais) ou couper (en français)  
qui supprime les blancs au début et à la fin



- La différence entre minuscules et majuscules

Solution :

convertir les mots/chaines de caractères en majuscules  
avant de les comparer



# V2A : RISQUES D'ERREUR



```
à phraseContientMots listeMotsATrouver phrase
résultat initialise local motsTrouves à vrai
dans
  faire
    si pas est une liste? objet obtenir listeMotsATrouver
    alors mettre listeMotsATrouver à diviser texte obtenir listeMotsATrouver
      à " ,"
    pour chaque mot dans la liste obtenir listeMotsATrouver
    faire
      si pas contient texte majuscule obtenir phrase
      sous-chaine majuscule couper obtenir mot
      alors mettre motsTrouves à faux
    résultat obtenir motsTrouves
```



```

initialise global numero à "0"
initialise global reponse à ""

quand Cadre1 . Touché
  x y touchedAnySprite
  faire
    mettre Screen1 . Image de fond à "BouleDeCristal2.jpg"
    appeler Reconnaissance_vocale1 . Obtenir texte
  
```

```

quand Reconnaissance_vocale1 . Après obtention texte
  résultat
  faire
    mettre Label1 . Texte à obtenir résultat
    mettre Screen1 . Image de fond à "BouleDeCristal.jpg"
    si contient texte obtenir résultat sous-chaine "éléphone"
    alors
      mettre global reponse à "Je ne sais pas"
      pour chaque sousListeMots dans la liste obtenir global ListeDesMots
      faire
        si phraseContientMots
          listeMotsATrouver obtenir sousListeMots
          phrase obtenir résultat
          alors
            mettre global numero à index dans la liste objet obtenir sousListeMots
            liste obtenir global ListeDesMots
            mettre global reponse à choisir liste élément liste obtenir global ListeDesRéponses
            index obtenir global numero
        sinon
          mettre global reponse à "tu ne mas pas dit le mot magique."
      appeler Texte_à_parole1 . Parler
      message obtenir global reponse
  
```

```

initialise global ListeDesMots à
  créer une liste
  créer une liste "la" "belle"
  créer une liste "le" "jeune"
  créer une liste "plus" "rapide"
  créer une liste "vieux"
  "bibliothèque"
  
```

```

initialise global ListeDesRéponses à
  créer une liste
  "Clara"
  "Lucas"
  "Emmanuel"
  "Lisa"
  "Pierre"
  "Marguerite Yourcenar"
  
```

```

à phraseContientMots listeMotsATrouver phrase
  résultat
  initialise local motsTrouves à vrai
  dans
    faire
      si pas est une liste? objet obtenir listeMotsATrouver
      alors
        mettre listeMotsATrouver à diviser texte obtenir listeMotsATrouver
        à ""
      pour chaque mot dans la liste obtenir listeMotsATrouver
      faire
        si pas contient texte majuscule obtenir phrase
        sous-chaine majuscule couper obtenir mot
        alors
          mettre motsTrouves à faux
      résultat
    obtenir motsTrouves
  
```