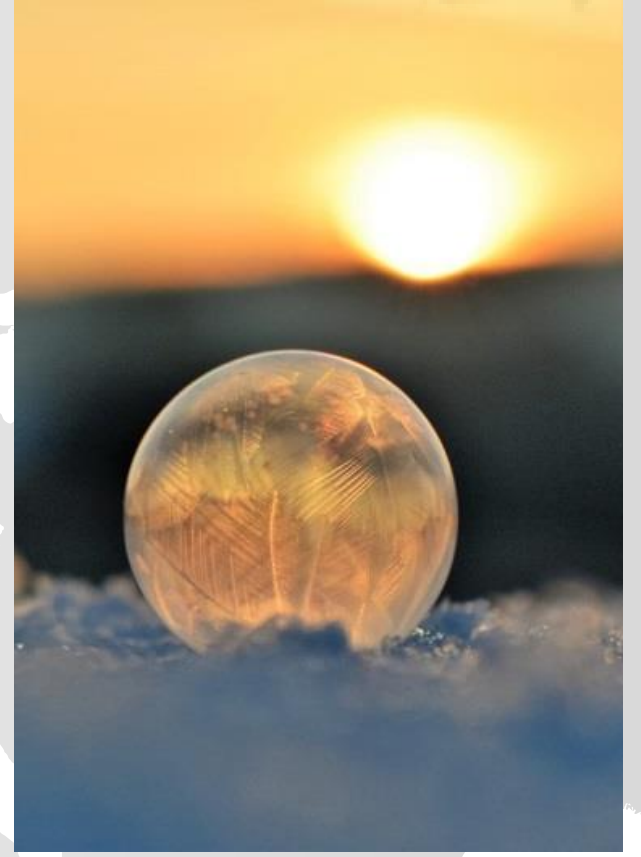


# App Inventor

enregistrer les  
données : tinyDB



Téléphone  
dis-moi qui est  
la plus belle ?



# ENREGISTREMENT DES DONNÉES



- Dans cette présentation nous allons voir comment :
  - Enregistrer avec **tinyDB** les données de l'application dans une base de données locales avec la liste des mots clefs et la liste des réponses.  
L'enregistrement des données sera fait après chaque modification :
- Relire ces données au démarrage de l'application

# ENREGISTREMENT DES DONNÉES



- Dans cette présentation nous allons voir comment :
  - **Enregistrer** avec **tinyDB** les données de l'application dans une base de données locales avec la liste des mots clefs et la liste des réponses.  
L'enregistrement des données sera fait après chaque modification :
    - à chaque ajout (dans les listes de mots et de réponses)
    - À chaque suppression (dans les listes de mots et de réponses)
  - **Relire** ces données au démarrage de l'application



# INTRODUCTION À TINYDB

- tinyDB est une base de données locale
  - elle permet d'enregistrer des données dans la mémoire de son téléphone
- C'est une base de données très simple
  - Elle enregistre un paquet de données sous une **étiquette** ou **tag**
  - Elle enregistre et relit les données sans s'occuper du contenu
  - C'est à l'application de se rappeler comment exploiter ce qu'il y a dedans
- Nous allons enregistrer les listes de mots et réponses
- Pour vous familiariser avec tinyDB regardez la vidéo à l'adresse

<https://onvaessayer.org/appinventor?video=tinyDB>



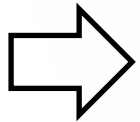
# ENREGISTREMENT DES DONNÉES

- Dans cette présentation nous allons voir comment :
  - Enregistrer avec **tinyDB** les données de l'application dans une base de données locales avec la liste des mots clefs et la liste des réponses.  
L'enregistrement des données sera fait après chaque modification :
    - à chaque ajout (dans les listes de mots et de réponses)
    - À chaque suppression (dans les listes de mots et de réponses)
  - **Relire** ces données au démarrage de l'application
- Les solutions sont dans la catégorie "stockage« des composants
  - fichiers
  - **tinyDB** (base de données locale)
  - tinyWebDB, ou firebase (base de données dans le cloud)



# 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>



# POINT DE DÉPART

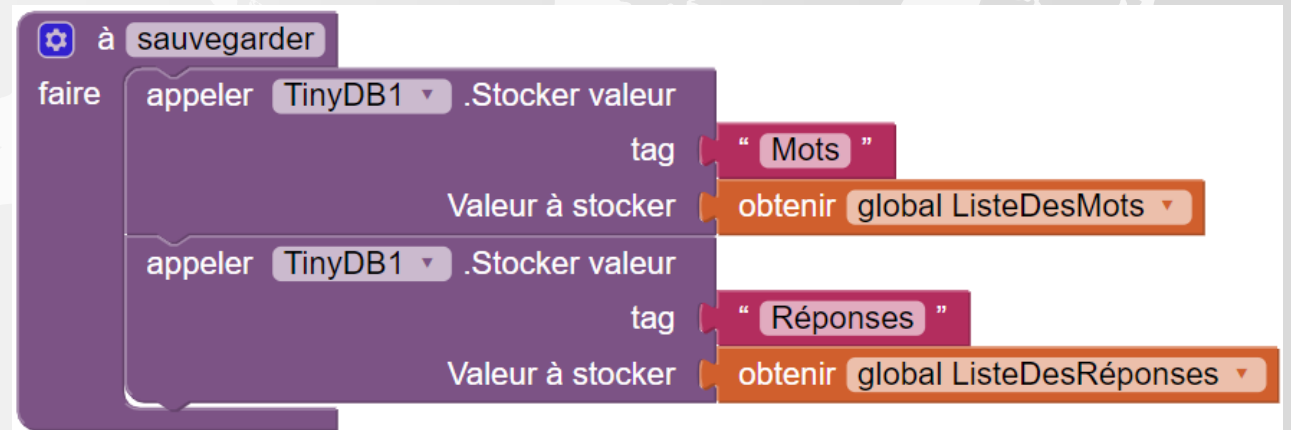


- L'enregistrement et la récupération de données deviennent intéressants à partir du moment où l'utilisateur peut les modifier et veut récupérer ses modifications.
- Vous pouvez repartir de la version 3 où vous avez appris à modifier ces listes.
  - Enregistrez cette version sous un nouveau nom
  - Et ajoutez un composant tinyDB depuis la partie design
- Ici nous allons :
  - Lire les listes dans tinyDB à chaque démarrage de l'appli (**screen1. Intialize**)
  - Enregistrer les listes après chque modification (ajout ou destruction)

# PROCÉDURE D'ENREGISTREMENT



- Dans "Qui est la plus belle" on veut enregistrer
  - la liste des mots clefs (ou liste de Liste si vous avez fait l'exercice)
  - la liste des réponses
- On va utiliser une procédure pour effectuer cet enregistrement
  - les listes de mots et réponses sont des variables globales,
  - on enregistre la **listeDesMots** sous l'étiquette **Mots**
  - et la **listeDesRéponses** sous l'étiquette **Réponses**





# APPELER CETTE PROCÉDURE ?

- La procédure d'enregistrement doit être appelée
  - après chaque modification : ajout ou suppression (ou à la sortie de l'application)
- Appel après chaque ajout :
  - Appel à la sauvegarde après les blocs de suppression dans les listes.

```
quand SélectionnerEtSupprimer .Après prise  
faire  
  supprimer liste élément liste obtenir global ListeDesMots  
                                index SélectionnerEtSupprimer . Indice sélection  
  supprimer liste élément liste obtenir global ListeDesRéponses  
                                index SélectionnerEtSupprimer . Indice sélection  
appeler sauvegarder
```

- Appel après chaque suppression :
  - dans le script de l'évènement **boutonAjouter.clic**  
appel à la procédure de sauvegarde après les blocs d'ajout dans les listes

```
sinon  
  ajouter éléments à la liste liste obtenir global ListeDesMots  
                                item Zone_de_texte_MotClef . Texte  
  ajouter éléments à la liste liste obtenir global ListeDesRéponses  
                                item Zone_de_texte_Reponse . Texte  
appeler sauvegarder
```

# PROCÉDURE DE LECTURE



- Pour lire les données dans la base, on peut définir une procédure
- L'appel à cette procédure se fait au démarrage de l'appli
- Code de la procédure
  - tinyDB.obtenir valeur est appelé pour chacune des listes
  - Si l'étiquette n'est pas trouvée, la procédure renvoie un liste vide.

à lireTinyDB  
faire

quand Screen1 .Initialise  
faire appeler lireTinyDB

à lireTinyDB  
faire  
mettre global ListeDesMots à appeler TinyDB1 .Obtenir valeur tag " Mots " Valeur si tag non présent créer une liste vide  
mettre global ListeDesMots à appeler TinyDB1 .Obtenir valeur tag " Réponses " Valeur si tag non présent créer une liste vide

- Attention à utiliser exactement la même étiquette (tag) qu'à l'écriture

# AMÉLIORATION 1



- Pour éviter d'avoir une liste vide la première fois
  - Au lieu de renvoyer une liste vide renvoyer la liste des valeurs initiales

```
à lireTinyDB
faire
  mettre global ListeDesMots à appeler TinyDB1 .Obtenir valeur
  tag " Mots "
  Valeur si tag non présent obtenir global ListeDesMots
  mettre global ListeDesMots à appeler TinyDB1 .Obtenir valeur
  tag " Réponses "
  Valeur si tag non présent obtenir global ListeDesRéponses
```

# AMÉLIORATION 2 (OPTION)



- Pour éviter les risques de divergence des listes de mots et de réponses qui sont enregistrées séparément
  - C'est mieux de regrouper toutes les données sous la même étiquette (tag)

```
à lireTinyDB
faire
  initialise local data à créer une liste
  créer une liste vide
  créer une liste vide
  dans
    mettre data à appeler TinyDB1 .Obtenir valeur
    tag "data"
    Valeur si tag non présent
    créer une liste
    obtenir global ListeDesMots
    obtenir global ListeDesRéponses
    mettre global ListeDesMots à choisir liste élément liste obtenir data
    index 1
    mettre global ListeDesRéponses à choisir liste élément liste obtenir data
    index 2
  à sauvegarder
  faire
    appeler TinyDB1 .Stocker valeur
    tag "data"
    Valeur à stocker
    créer une liste
    obtenir global ListeDesMots
    obtenir global ListeDesRéponses
```