

# Visual Basic pour Applications

Les interfaces graphiques

**Romain Tavenard**  
**Kévin Huguenin**  
**Christophe Avenel**

*Romain.Tavenard@irisa.fr*

*Kevin.Huguenin@gmail.com*

*Christophe.Avenel@irisa.fr*

20 Novembre 2008

# Outline

## 1 Retour sur les objets d'Access

## 2 Les interfaces graphiques

- Les interfaces graphiques simples
- Les interfaces graphiques plus élaborées

# Liste des objets à manipuler

Nous avons introduit les objets suivants :

- DAO.Database et sa méthode OpenRecordset
- DAO.Recordset et ses champs :
  - Fields
  - EOF
  - MoveNext
- DAO.Field et ses champs :
  - Name
  - Value

Cas particulier des champs vides :

- Utilisation de IsNull plutôt que == ""

# Outline

## 1 Retour sur les objets d'Access

## 2 Les interfaces graphiques

- Les interfaces graphiques simples
- Les interfaces graphiques plus élaborées

# InputBox et MsgBox

```
InputBox(prompt[, title] [, default] [, xpos] [, ypos] [,  
helpfile, context])
```

```
MsgBox(prompt[, buttons] [, title] [, helpfile, context])
```

# Un exemple d'interface graphique plus élaboré

The screenshot shows a VBA dialog box titled "Listes simples". It features two text input fields at the top labeled "Nom:" and "Prénom:". Below these is a list box containing three entries: "Huguenin, Kévin", "Tavenard, Romain", and "Avenel, Christophe". The first entry is selected. To the right of the list box are three buttons: "Ajouter", "Supprimer", and "Vider". At the bottom of the dialog are "Ok" and "Quitter" buttons.

Nom	Prénom
Huguenin, Kévin	
Tavenard, Romain	
Avenel, Christophe	

# Dessiner une interface graphique dans Visual Basic

On commence par créer un UserForm ("Insertion" → "UserForm").  
Puis on y ajoute des contrôles à l'aide de la boîte à outils :



# Grands principes

Comme souvent en Visual Basic, *tout est objet* :

- Fenêtres
- Boutons
- Zones de texte
- etc.

Programmer une interface graphique, c'est :

- 1 Prévoir les actions de l'utilisateur
- 2 Comprendre les événements qui y sont associés
- 3 Définir le comportement du programme lorsqu'un de ces événements arrive



# Programmation événementielle

Événements typiques :

- Clic sur la croix pour fermer une boîte de dialogue
- Clic sur un bouton
- Double-clic sur un objet
- etc.

Pour connaître les événements associés à un objet, il suffit de consulter l'Aide de VB.

# Programmation événementielle et bonnes pratiques

Voici une liste succincte (mais primordiale) de bonnes pratiques :

- Limiter au maximum le code associé aux UserForm (utiliser des procédures définies dans les modules)
- À la fermeture d'un UserForm, stocker les valeur entrées dans des variables globales
- Cas particulier d'une suite de fenêtres :
  - Gérer l'enchaînement par un module externe (UserFormX.Show)
  - Différences entre Terminate et QueryClose
  - Fermeture d'une fenêtre par Unload Me