

Activité 6 : Excel et Word

Utilisez le même fichier Excel [td_excel_1.xls](#) contenant le modèle de facture.

Introduction à la programmation objet VBA

Dans Excel, la programmation VBA permet de **piloter** d'autres applications Microsoft et échanger des données.

Travail à faire

Lancer Word depuis Excel

Dans le classeur Excel **tdexcel1.xls** et écrivez la procédure **LancerWord()** dans le module **DCG**. Puis ajoutez un bouton permettant de lancer cette procédure.

Code VBA à utiliser :

```
Set WordApp = CreateObject("word.application")
WordApp.Visible = True
```

Exécutez la procédure **LancerWord()** à partir de votre bouton et vous verrez que Word se lance mais sans aucun document.

Créer un document Word

Complétez votre procédure **LancerWord()** pour créer un document Word avec les informations suivantes :

- mettre un titre **Confirmation Achat**
- Titre : centré
- Style de police : gras
- Taille de police : 20

Code VBA à utiliser :

```
Set docWord = WordApp.Documents.Add
With docWord.Sections(1).Range
    .Text = "Confirmation Achat" & Chr(10)
    .Font.Bold = True
    .Font.Size = 20
    .ParagraphFormat.Alignment = wdAlignParagraphCenter
End With
```

Le caractère **&** permet de faire des concaténations de chaînes de caractères.

L'instruction **Chr(10)** permet de gérer un saut de ligne. Il est possible d'utiliser des constantes VBA ce qui est plus facile à utiliser. <http://www.commentcamarche.net/contents/1165-vbscript-les-constantes> [Les constantes VB]

La notation **With ... End With** permet de simplifier l'écriture des commandes qui concernent le même objet. Sinon il aurait fallu écrire :

```
Set docWord = WordApp.Documents.Add
docWord.Sections(1).Range.Text = "Confirmation Achat" & Chr(10)
docWord.Sections(1).Range.Font.Bold = True
docWord.Sections(1).Range.Font.Size = 20
docWord.Sections(1).Range.ParagraphFormat.Alignment = wdAlignParagraphCenter
```

Enregistrer le document Word

Complétez votre procédure **LancerWord()** pour enregistrer votre document Word nouvellement créé :

Code VBA à utiliser :

```
Dim strchemin As String
strchemin = ThisWorkbook.Path & "\"
With docWord
    .SaveAs (strchemin & "ConfirmationFacture.docx")
    .Close savechanges:=wdDoNotSaveChanges
End With
```

Fermer le document Word

Pour fermer le document Word il suffit de quitter le logiciel et de détruire les variables objets qui ont été créées :

Code VBA à utiliser :

```
WordApp.Quit
Set docWord = Nothing
Set WordApp= Nothing
MsgBox strchemin & "ConfirmationFacture.docx"
```

Retour au menu de Excel & VBA

[Excel et VBA \(Visual Basic pour Application\)](#)

From:

[/ - Les cours du BTS SIO](#)

Permanent link:

[/doku.php/dev/excelvba/td6excelword](#)

Last update: **2016/01/20 13:54**

