

Checklist des Meilleures pratiques en développement VBA

Auteur : Philippe Heinrich

Site web : heinrich-consultant.fr

E-Mail : ph@heinrich-consultant.fr

Voici la checklist (pour plus de détails voir <https://www.heinrich-consultant.fr/vba-et-scripts/les-meilleures-pratiques-du-developpement-en-vba/>)

Environnement

- Je m'interroge sur les données et la pertinence de VBA
- Je rends l'application le plus possible indépendante de l'environnement
- Je documente mon travail, j'indente et j'aère mon code
- Je réalise des tests unitaires (pour les applications importantes)
- J'utilise un outil de gestion des versions (pour les applications importantes)
- Composants COM : je me pose la question de la liaison anticipée ou de la liaison tardive
- Je rends les fonctions indépendantes pour pouvoir les réutiliser

Variables

- Forcer la déclaration de variables avec Option Explicit, typer les variables et les déclarer sur la même ligne si elles sont reliées entre elle
- Adopter une convention de nommage des variables, par exemple la notation hongroise et le Camel Case
- Déclarer les variables près de l'endroit où elles sont utilisées
- Ne pas utiliser le type Variant sans une bonne raison
- Utiliser les constantes et énumérations VBA, et déclarer vos propres énumérations quand c'est possible
- Utiliser des modules de classe quand c'est justifié
- Vider les variables objets après utilisation

Procédures et fonctions

- Nommer correctement ses procédures et fonctions
- Limiter ses procédures et fonctions à 30 lignes
- Utiliser judicieusement ByVal et ByRef pour le passage d'arguments
- Vérifier le type et la valeur des arguments passés à une procédure ou fonction
- Utiliser des Procédures/Fonctions dédiées pour le code qui pourrait retourner une erreur à l'exécution
- Ne pas permettre à l'utilisateur de lancer des sous-programmes qui ne sont pas conçus pour.
- Utiliser « Exit Do » et « Exit For » avec prudence
- Eviter d'utiliser l'instruction Goto, sauf pour le traitement d'erreur

Accès aux objets de MS Excel

- Préciser quel objet vous voulez manipuler et le manipuler directement
- Manipuler des arrays (tableaux de variables) plutôt que des plages de cellules
- Désactiver certaines fonctionnalités pour accélérer le développement
- Eviter de copier/coller des cellules
- Utiliser la syntaxe With/End With
- Préférer utiliser WorksheetFunction à une procédure personnalisée

Traitement des erreurs

- Mettre en place un traitement d'erreurs
- Compiler avant de lancer le programme
- Utiliser judicieusement les instructions Try...Catch...Finally
- Sauvegarder les erreurs et leur contexte dans un fichier texte