



ORGANISME DE FORMATION AUX TECHNOLOGIES ET METIERS DE L'INFORMATIQUE

Certification TOSA VBA *par l'éditeur TOSA Isograd*

No ACTIVITE : 11 92 18558 92

TELEPHONE : 01 85 77 07 07

E-MAIL : inscription@hubformation.com

Passage de l'examen

L'examen se passe en ligne, à la date de votre choix, sous e-surveillance.
Vous devez disposer d'une webcam, ouvrir votre microphone et être dans un environnement silencieux.
Vous recevrez vos accès pour vous connecter à votre espace candidat.

Référence	TOSA-VBA
Certificateur	TOSA Isograd
Code RS	6963
Tarif	90 €HT

[ALLER SUR NOTRE SITE](#)

Détails de l'examen

Cet examen est référencé au répertoire spécifique de France Compétences sous le code RS6963 et permet un financement CPF pour la/les formation(s) y préparant.

Automatiser des processus dans les applications Microsoft Office avec VBA (Tosa)

L'examen TOSA se passe en centre agréé ou en ligne. Votre score (sur 1000) vous permettra de communiquer sur votre niveau de connaissances du langage.

| Durée : 1h

| 25 questions QCM et cas pratiques

| Score minimum : 551

| Version VBA Excel 2019

| Date d'enregistrement de la certification : 19/12/2024

Domaine 1 : Gestion des objets VBA

1.1 Manipuler les objets Excel : automatiser les processus dans Excel en utilisant et en gérant les principaux objets tels que Workbooks (cahiers de travail), Worksheets (feuilles de calcul), Ranges (plages de cellules), et Charts (graphiques), afin de simplifier les tâches spécifiques et optimiser la gestion des données.

1.2 Gérer les collections d'objets: gérer les groupes d'objets similaires dans Excel, comme les collections Worksheets ou Cells, en parcourant, ajoutant ou supprimant des éléments de la collection, et en utilisant des boucles afin d'automatiser les opérations répétitives sur chaque élément de la collection.

1.3 Interfacer avec d'autres applications: utiliser VBA pour interagir avec d'autres programmes Microsoft 365 tels que Word, Access et Outlook, afin d'automatiser des tâches inter-applications, comme l'exportation de données depuis Excel vers Word ou l'envoi d'emails via Outlook, en exploitant les objets COM (Component Object Model) pour fluidifier les processus.

Domaine 2 : Procédures VBA

2.1 Écrire des procédures : créer des sous-routines et des fonctions en VBA pour exécuter des tâches spécifiques, en utilisant des arguments et des paramètres pour passer des données entre les procédures, afin d'améliorer la modularité et la réutilisabilité du code.

2.2 Traiter les erreurs et exceptions : mettre en place des mécanismes de gestion des erreurs en utilisant des structures et des blocs conditionnels pour anticiper et traiter les erreurs, et en développant des procédures de nettoyage et de récupération après une erreur, afin de rendre le code plus robuste et moins sujet aux plantages.

2.3 Améliorer la performance du code : optimiser les routines VBA en minimisant les accès aux objets Excel, en utilisant des types de données appropriés, en

optimisant les boucles et en gérant la mémoire de manière efficace, afin qu'elles s'exécutent plus rapidement et de manière plus performante.

Domaine 3 : Boîtes de dialogue, formulaires et contrôles

3.1 Concevoir des formulaires : créer des UserForms (formulaires utilisateur) en VBA en configurant l'interface utilisateur, en ajoutant des contrôles tels que des boutons, des étiquettes et des zones de texte, et en gérant leur disposition ainsi que leurs propriétés, afin de faciliter l'interaction avec l'utilisateur.

3.2 Implémenter des contrôles ActiveX : ajouter et gérer des contrôles ActiveX dans les formulaires ou directement dans les feuilles de calcul Excel, en insérant des contrôles tels que des boutons, des cases à cocher et des listes déroulantes, afin d'améliorer l'interactivité et la fonctionnalité des applications.

3.3 Gérer les événements de formulaires : écrire du code pour répondre aux actions de l'utilisateur sur les formulaires, en utilisant les événements associés aux contrôles et aux formulaires, afin de déclencher des actions spécifiques et d'améliorer l'interactivité des applications.

Domaine 4 : Environnement VBA et utilisation des outils de débogage

4.1 Utiliser les outils de débogage : connaître et utiliser les outils intégrés dans l'IDE VBA, tels que les points d'arrêt, la fenêtre d'exécution, la surveillance des variables et l'exécution pas à pas du code, afin d'identifier et de corriger les erreurs efficacement.

4.2 Analyser et suivre le code : utiliser les fonctionnalités de traçage et de journalisation en employant des instructions Debug.Print pour afficher des messages dans la fenêtre Exécution et en analysant les flux d'exécution, afin de comprendre le comportement du code, identifier les problèmes potentiels et optimiser les routines.

4.3 Optimiser l'environnement de développement : configurer l'IDE VBA en personnalisant l'éditeur, en gérant les références de projet, en utilisant des modèles de code et en mettant en place un environnement structuré, afin de maximiser l'efficacité du développement et faciliter la gestion des projets VBA.

Accessibilité



Pour tout besoin spécifique (vue, audition...), veuillez nous contacter au 01 85 77 07 07.