Accueil > Bureautique > Access > Access Avancé > Protéger l'accès à un formulaire Access





Pour partager cette vidéo sur les réseaux sociaux ou sur un site, voici son url : https://youtu.be/7LNmdyqCSMs?list=PLpQBnWleLAas1yqMbmVGJl1WHX-z-jhQs



Sujets que vous pourriez aussi aimer :









Protéger l'accès à un formulaire

Dans cette formation Access, nous apportons la solution pour sécuriser l'exécution des formulaires sensibles. Certains de ces outils sont en effet destinés à intervenir sur les données de la base. Il peut par exemple s'agir d'un formulaire de suppression ou d'un formulaire de mise à jour. Ces instruments ne peuvent être mis entre toutes les mains. Leurs conséquences peuvent s'avérer désastreuses car irréversibles. L'idée est de verrouiller l'accès à ces formulaires par un mot de passe administrateur.

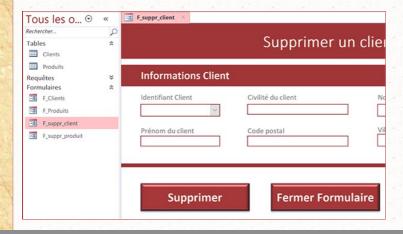
Base de données source

Pour la mise en place de la solution dans les meilleures conditions, nous proposons de réaliser les travaux à partir d'une base de données offrant des outils fonctionnels.

- Télécharger la base actions-clients-et-commandes.accdb en cliquant sur son lien,
- Double cliquer sur le fichier téléchargé pour l'ouvrir dans Access,
- Puis, cliquer sur le bouton Activer le contenu du bandeau de sécurité,

Le volet des objets Access s'affiche sur la gauche de l'écran. Il énumère les outils disponibles à ce stade. On compte deux tables et quatre formulaires. La table Clients recense les clients de l'entreprise. La table Produits énumère tous les articles en vente. Les formulaires F_Clients et F_Produits sont conçus sur ces tables respectives. Ils permettent la consultation et la modification des informations.

Les **formulaires F_suppr_client** et **F_suppr_produit** sont particuliers et sensibles. Ils doivent permettre de purger des enregistrements des tables source, sur désignation d'un identifiant ou d'une référence.



Etant données les conséquences qu'ils engendrent, il est absolument nécessaire de protéger leur accès par mot de passe. Seuls les utilisateurs avertis, comme les administrateurs, doivent avoir l'autorisation d'intervenir sur le coeur des données.

Pour ce type d'application, il serait judicieux d'offrir un formulaire d'accueil constitué de boutons. Ces derniers permettraient de détourner l'utilisateur sur les différents outils de la base de données. C'est une solution que nous proposerons dans une prochaine formation.

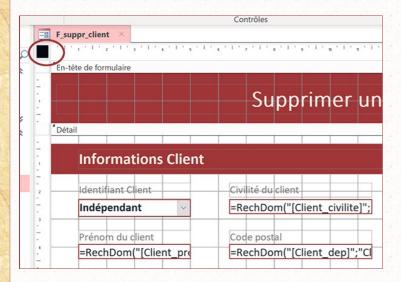
Et dans ces conditions, il apparaît légitime de personnaliser l'environnement de manière à <u>restreindre les possibilités</u>. Nous avions démontré ces techniques et leurs limites. Bientôt, nous comprendrons comment il est possible de gérer tous les droits.

Ouverture du formulaire par mot de passe

L'enjeu de cette formation est de vérifier et valider un **mot de passe** en contournant le **code VBA**. Une **macro** déclenchée à l'ouverture du **formulaire** est une solution. Elle permet d'analyser des conditions comme la <u>fonction Si d'Excel</u>. De plus, le **générateur d'expression** met à disposition toutes les **fonctions d'Access** qu'il est donc possible d'imbriquer. L'une d'entre elles propose d'interroger l'utilisateur par le biais d'une boîte de dialogue. C'est donc la réponse fournie qui doit être étudiée pour analyser le fameux mot de passe, décidant de l'ouverture ou de la fermeture du formulaire.

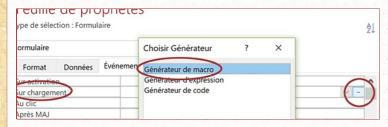
- Dans le volet des objets Access, cliquer droit sur le formulaire F_suppr_client,
- Dans la liste, choisir Mode création,

Nous basculons ainsi dans la vue en conception du formulaire.



La feuille de propriétés doit être disponible. Et, c'est le bouton du même nom dans le ruban contextuel Création qui permet de l'afficher ou de la masquer. Elle permet d'ajuster les attributs du contrôle actif sur le formulaire. Et comme l'indique le carré noir, en haut à gauche et à l'intersection des règles, c'est naturellement le formulaire lui-même qui est actif par défaut. C'est bien lors de son chargement en effet, que nous souhaitons intervenir pour verrouiller l'accès.

- Dans la feuille de propriétés, activer l'onglet Evénement,
- Puis, cliquer sur le petit bouton à l'extrémité de son événement Sur chargement,
- Dans la boîte de dialogue qui suit, choisir **Générateur de macro** et valider par Ok,



Nous basculons ainsi dans l'éditeur de macro. Une liste déroulante permet d'empiler les actions à engager, les unes sous les autres. Mais ici, l'action est unique et elle est soumise à condition. Si le mot de passe n'est pas correct, le formulaire doit être fermé.

• Dans le volet du catalogue d'actions, double cliquer sur la fonction Si,



Elle appartient à la catégorie Déroulement de programme. Une zone de saisie l'accompagne au centre de l'éditeur. Il s'agit de construire le critère à vérifier. Celui-ci consiste à comparer le mot de passe demandé à celui saisi par l'utilisateur tentant d'ouvrir le formulaire. Pour cela, nous devons déclencher une boîte de dialogue proposant une zone de saisie. Et nous devons comparer le résultat retourné à celui du mot de passe nécessaire.

• A l'extrémité droite de la zone du Si, cliquer sur l'icône de la petite baguette magique,

Cette action a pour effet de déclencher l'affichage du générateur d'expression. C'est grâce à lui que nous allons trouver la fonction Access offrant une boîte de message permettant d'interagir avec l'utilisateur.

- Dans la liste de gauche, déployer la catégorie Fonction,
- · Puis, sélectionner l'élément Fonctions intégrées,
- Dans la liste du centre, sélectionner la catégorie Messages,
- Dans la liste de droite, double cliquer sur la fonction BEntrée,

Sa syntaxe s'inscrit ainsi automatiquement en haut du générateur d'expression :

BEntrée(«prompt»; «title»; «default»; «xpos»; «ypos»; «fichier_aide»; «contexte»)

7 arguments sont proposés. Mais seul le premier est nécessaire. Il s'agit du message adressé à l'utilisateur pour lui indiquer la réponse à fournir. Nous pouvons éventuellement renseigner le deuxième argument. Il s'agit du titre de la boîte de dialogue.

• A la place de l'argument «Prompt», saisir le message suivant entre guillemets :

'Veuillez saisir le mot de passe d'ouverture'

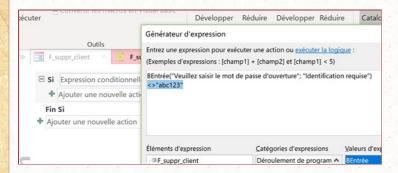
• A la place de l'argument «title», saisir le titre suivant entre guillemets :

'Identification requise'

• Puis, supprimer tous les arguments qui suivent avec les points-virgules qui les accompagnent,

L'expression n'est pas encore terminée. A ce stade, la fonction doit déclencher une boîte de dialogue à destination de l'utilisateur. Et comme vous le savez, par définition, après son traitement, une fonction retourne une valeur à étudier. Cette valeur est l'information saisie par l'utilisateur. Nous devons la comparer avec le mot de passe requis pour accéder au formulaire. Arbitrairement, nous décidons que ce mot de passe est la suite de caractères : abc123.

• En haut de l'éditeur, terminer l'expression avec l'inégalité suivante : <> 'abc123',



La question consiste donc à savoir si le mot de passe renseigné par l'utilisateur est différent de celui demandé, soit : abc123. Dans ces conditions, le formulaire doit être fermé car l'accès est refusé. Dans le cas contraire, le mot de passe étant conforme, aucune action n'est requise dans la mesure où l'ouverture du formulaire doit se poursuivre.

• Cliquer sur le bouton Ok pour valider l'expression,

Nous sommes de retour sur l'éditeur de macro. La syntaxe est effectivement inscrite dans la zone de critère de la fonction Si.

- Sous l'expression de la condition, déployer la liste déroulante,
- Dans la liste des suggestions, choisir l'action FermerFenêtre,
- Dans la zone Type d'objet, choisir Formulaire,
- Dans la zone Nom d'objet, désigner le formulaire F_suppr_client,
- · Dans la zone Enregistrer, choisir Non,

Donc, lorsque le mot depasse n'est pas valide, nous fermons le formulaire en cours d'ouverture. En d'autres termes, nous refusons l'accès. Comme nous le disions, nous pourrions enclencher des actions dans le cas contraire, soit lorsque le mot de passe est conforme.





Et pour cela, il suffit de cliquer sur le lien Ajouter Sinon en bas à droite de l'action Si. Dès lors, une nouvelle liste déroulante se propose pour imbriquer les actions correspondantes à enclencher. Mais, nous le savons, lorsque le mot de passe est correctement saisi, nous souhaitons simplement laisser se dérouler l'accès au formulaire











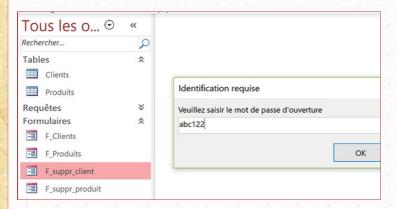
- Dans le ruban Création en haut de la fenêtre, cliquer sur le bouton Enregistrer,
- Puis, cliquer sur le bouton Fermer,

De la sorte, nous sommes de retour sur le formulaire en mode conception.

• Cliquer sur le symbole de la disquette dans la barre d'accès rapide, en haut à gauche de la fenêtre,

Ainsi, nous mettons à jour les modifications opérées sur le formulaire, notamment pour la liaison avec la macro que nous venons de finir de configurer. Vous pouvez aussi exploiter le raccourci clavier CTRL + S.

- Cliquer sur la croix de l'onglet pour fermer le formulaire,
- Puis, dans le volet des objets Access, double cliquer de nouveau sur son nom pour l'exécuter,



Cette fois, grâce à la fonction BEntrée, une boîte de dialogue munie d'une zone de saisie se déclenche en effet. Si vous entrez un nouveau mot de passe, l'écran reste vierge grâce à la fonction FermerFenêtre déclenchée par la branche Alors de l'action de macro Si. Dans le cas contraire, l'accès est autorisé et le formulaire apparaît. Nous aurions éventuellement pu cumuler une seconde action dans la branche Alors de la fonction Si. Elle aurait consisté à afficher une boîte de message explicative avec un message explicite : Mot de passe erroné, accès interdit. C'est l'action de macro ZoneMessage qui permet l'affichage d'une telle boîte démunie de zone de saisie.

Quoiqu'il en soit, nous avons parfaitement réussi à sécuriser l'accès à ce formulaire sensible grâce à une action de macro. Cette dernière vérifie une condition sur la saisie utilisateur au chargement du formulaire. C'est exactement le même processus qu'il s'agit de reproduire pour sécuriser l'accès au formulaire F_suppr_produit.

🖈 Le saviez-vous ? Dans une table Access, le raccourci clavier CTRL+ ' permet de répliquer l'information du champ situé juste au-dessus













