

Vocabulaire et structure d'une base de données



Reprenons notre base Access. Dès sa création, avant même de saisir la moindre donnée, on peut se rendre compte qu'elle est constituée de plusieurs éléments :
[tables](#), [requêtes](#), [formulaires](#), [états](#), [macros](#), [modules](#).

Voilà les premiers mots typiques quand on travaille sur des bases de données!

Les tables tout d'abord.

C'est dans une table que vous allez enregistrer vos informations. Par information, on entend ici vos données.

■ Vous commencez par définir vos **champs**. C'est l'étape la plus importante.

- le **nom** du champ correspond à l'en-tête de colonne de votre tableau ;
- il est associé à un **type de données**, qui définit le format des données saisies dans cette colonne.

Comme type de données, vous retrouvez en particulier :

<i>Type de données</i>	<i>Définition</i>	<i>Exemples d'utilisation</i>
Texte	chaîne alphanumérique limitée à 255 caractères	Nom, Adresse, Profession, Titre, Téléphone, ...
Mémo	bloc de texte pouvant contenir jusqu'à 65536 caractères	Résumés, Informations, CV, ...
Numérique	caractères numériques, entiers ou non (le séparateur - point ou virgule - est celui défini dans les paramètres régionaux de votre panneau de configuration)	Age, Nombre d'enfants, ...
Date/Heure	données au format date (03/07/2000 par exemple)	Date d'inscription, de naissance, heure d'envoi d'un message, ...
NuméroAuto	nombre qui s'incrémente automatiquement à chaque nouvel enregistrement dans la table	Référence client, Classement, ...

■ Ensuite, vous pouvez **saisir vos données**, directement dans la table. Là vient la différence de "fonctionnement" entre tableur et base de données!

- à chaque cellule que vous remplissez, vous saisissez une **donnée** ;
- à chaque ligne que vous entrez, correspond un **enregistrement** : c'est une fiche qui contient des renseignements (données pour chaque champ) sur un élément unique de la liste ;
- chaque ligne doit être unique, et différenciable des autres : c'est à cela que sert la **clef primaire**. La clef primaire d'une table correspond au champ pour lequel toutes les données sont différentes d'un enregistrement à l'autre.

exemple :

- si vous avez un champ "Nom de famille", et que tous les noms sont différents, alors ce champ peut servir de **clef primaire** ;
- par contre si vous avez des homonymes dans la liste, vous pouvez retrouver deux enregistrements pour deux personnes différentes, mais avec une même valeur "nom de famille" - là il faut trouver une autre **clef primaire** ;
- c'est à cela que sert le **NuméroAuto** : en utilisant ce type de données, on est sûr de travailler avec au

moins un champ sans doublon.

- enfin, tous vos champs et vos enregistrements constituent une **table** ;
- toutes les tables composent une seule **base de données**.

■ Enfin, vos tables ont souvent des points communs entre-elles. Si c'est le cas, pour éviter les redondances vous allez **créer des relations**. Nous n'en avons pas besoin pour débiter l'ASP, mais il est important d'en comprendre la logique.

- vous avez une table "clients", qui contient la référence du client, son nom, et son adresse ;
- une deuxième table, "commande", qui contient la référence de la commande, celle du client, la date de la commande.

Pour savoir à quelle adresse envoyer la facture de la commande, vous avez besoin de préciser que la référence du client dans une table correspond à sa référence dans l'autre table.

Démarche :

1. je prends la référence du client dans la table "commandes" pour la ref. de la commande qui m'intéresse ;
2. je vais chercher dans la table "clients" l'enregistrement qui correspond à cette référence ;
3. je récupère le Nom et l'adresse du client qui correspond.

Remarque :

En ASP, l'utilisation de la condition **"Where"** supplée à la création de cette relation (voir Module 2).



Les Requêtes.

■ Quand on travaille sur les bases de données, on l'a vu, les données sont enregistrées, brutes, dans des tables. **Pour y avoir accès**, pour en sélectionner quelques unes ou effectuer des calculs, on doit passer par des requêtes.

- une requête **enregistre vos critères** de sélection, ou les calculs que vous souhaitez réaliser : ainsi, à chaque fois que vous exécutez la requête, elle prend en compte les dernières informations saisies dans la table, sans y modifier aucune valeur (sauf si vous le demandez explicitement, mais nous n'aborderons pas le sujet ici) ;
- **exécuter une requête**, c'est visualiser toute ou une partie de votre table, à un instant t, sous certaines conditions que vous définissez vous-mêmes.

■ Si les requêtes sont typiques des bases de données, elles sont aussi directement **programmables en ASP**. Vous pourrez, par exemple, demander l'affichage de tous les clients habitant Paris ou les effacer de votre table, et ce en écrivant votre code ASP.



Les Formulaires.

■ Un formulaire permet de **faciliter la saisie** des données.

- un formulaire, c'est une **interface** plus ou moins soignée entre un utilisateur et la base : il permet de remplir les tables sans être directement sur la feuille de données ;

- réaliser un formulaire, c'est aussi **améliorer la saisie** des données (bulles d'aide, cellules mieux paramétrées pour recevoir les informations, ...).

■ L'ASP étant directement intégré à la page HTML, ce formulaire c'est **en HTML** et directement sur votre page que vous le réaliserez.

- **création** d'un formulaire en HTML ;
- **envoi des réponses** à une page ASP qui va les traiter (il s'agit le plus souvent d'un traitement pour que les données saisies dans les différents champs viennent s'inscrire dans la table).



Les Etats, Macros, Modules.

Sous Access, vous retrouvez ces trois autres formulations : les Etats, Macros, et Modules. Nous n'en avons pas besoin ici, mais nous les évoquerons tout de même...

■ **Un état** : c'est l'équivalent d'un aperçu avant impression.

- un peu un mélange entre une requête et un formulaire, le tout optimisé pour l'**impression** ;
- vous y définissez ce que vous voulez imprimer, et comment vous voulez l'imprimer, le logiciel vous aide en vous permettant d'**intervenir** sur la présentation, la sélection des informations. Vous pouvez même effectuer des calculs directement sur votre page.

■ **Une macro** : facilite un travail répétitif.

- une macro est une fonction qui vous permet d'**enregistrer une série d'opérations** ;
- à chaque **exécution** de la macro, ces mêmes commandes s'exécutent, vous permettant gain de temps et d'énergie! Vous pouvez même redéfinir votre environnement de travail (barres d'outils, ...) à partir de macros.

■ **Un module** : c'est un programme en SQL.

- vous **programmez** vous-même ces commandes en **SQL** sur une feuille qui peut intégrer quelques assistants.



N'hésitez pas à [nous contacter](#) si certaines parties vous semblent floues, ou si vous n'avez pas trouvé l'information recherchée... Ces pages sont sans cesse remises à jour, et vous y contribuez!



www.asp-irine.com