

Système d'information

Chapitre 1 : MS Access Niveau 1



Konan Marcellin BROU

marcellin.brou@inphb.ci

2024-2025

1

Sommaire

- **Introduction**
- **Généralités sur les BD**
- **Modèle Entité/Association**
- **Modèle relationnel**
- **SGBD Access**
- **Les tables**
- **Les requêtes**
- **Les formulaires**
- **Les états**
- **Bibliographie**

2

1

I. Introduction

□ Objectifs :

- Comprendre le fonctionnement du SGBD MS Access
- Savoir créer les différents objets d'Access (Table, Requêtes, Formulaire, Etat)

I. Introduction

□ 1.1. Présentation

■ Système

- Ensemble d'éléments matériels ou immatériels (homme, machine, méthode...) en interaction transformant par un processus des éléments (données d'entrées) en d'autres éléments (données de sortie).
- Un tel système est appelé système physique ou système opérant (SO).



□ Exemple 1 : une chaudière (SO)

- transforme par combustion du charbon (donnée d'entrée) en chaleur (donnée de sortie).



□ Exemple 2 : commande des clients



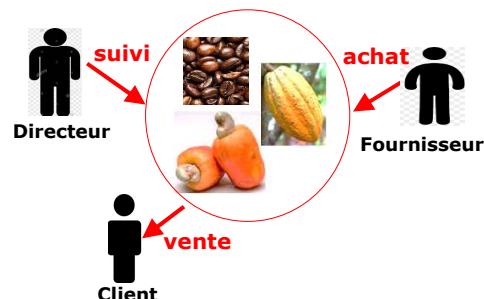
I. Introduction

■ Organisation

- **Entreprise, administration ou tout groupe social organisé exerçant une activité.**
- **Peut-être modélisé comme comportant trois sous-systèmes :**
 - **Système de pilotage** (celui qui réfléchit, décide, oriente).
 - **Système d'information** (données).
 - **Système opérant** (celui qui produit, transforme, agit).

□ Exemple : Société de vente de produits agricoles

- Achat de produits à des fournisseurs
- Vente des produits à des clients



I. Introduction

□ Différentes activités :

- Achat de produits à un fournisseur
- Vente de produit à un client,
- Suivi de l'évolution des stocks

□ Informations ou données

- Identification des produits : prix de vente, d'achat et quantités.
- Coordonnées des fournisseurs et des clients, etc.

■ Données brutes

- Collection d'éléments de valeur brute ou de faits servant à calculer, raisonner et mesurer ;
- Peuvent être collectées, stockées ou traitées;
- Ne possèdent pas de contexte ni de sens.
- Exemples :
 - !
 - -----

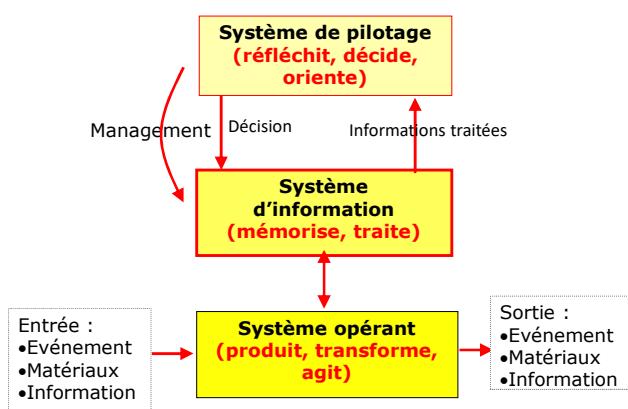
I. Introduction

- **Information = donnée + sens attaché à la donnée :**
 - Proviennent de l'organisation des données, mettent en valeur les relations entre les différents éléments de ces données ;
 - Fournissent un contexte et un sens aux données.
 - **Exemples :**
 - ! : point d'exclamation
 - : SOS (Save Our Soals)

- **Connaissance = information + mode d'emploi pour entreprendre une action**
 - Ce que l'on a appris par l'étude ou la pratique.
 - Vient de la compréhension de l'information dans son contexte ;
 - Utile au processus de décision.
 - **Exemples :**
 - Ecrire un "!" pour marquer une exclamation en fin de phrase,
 - Si le signal "...." est reçu alors déclencher l'alerte et envoyer des secours.

I. Introduction

- **Système d'Information (SI)**
 - Constitué de moyens humains, matériels et des méthodes permettant de collecter, mémoriser, traiter, distribuer, consulter les informations dont l'entreprise a besoin pour ses différents domaines d'activité.
 - **Moyens humains** : toutes les personnes touchées par le SI ;
 - **Moyen matériel** : machines, ordinateurs, supports d'information (disque, CD...) ;
 - **Méthode** : algorithme, programme, modèle mathématique, plan comptable...



I. Introduction

□ Un SI comporte deux parties :

- SI Organisationnel (SIO) : partie visible, basée essentiellement sur des ressources humaines, de l'activité organisée ;
- SI Informatisé (SII) : contenu informatisé du SI (logiciel, BD).

□ Le SI possède deux aspects :

- **Aspect statique** (ou aspect données) :
 - Base d'information,
 - Modèle (ou structure) de données.
- **Aspect dynamique** (ou aspect traitement) :
 - Circulation de l'information entre les différents acteurs.
 - Evolution chronologique des opérations provoquées par des événements.

II. Généralités sur les BD

□ 2.1. Définition d'une BD

- Ensemble (structuré, cohérent, intégré, partagé) des informations nécessaires au fonctionnement d'une entreprise.
- La gestion de ces informations est assurée par le Système de Gestion de Base de Données (SGBD).

■ Pourquoi les BD ?

□ Insuffisances des (SGF) :

- Redondance des informations, saisies multiples et incohérence des informations ;

□ Progrès technologiques

- Stockage de données de plus en plus volumineuses ;

□ Besoins des entreprises

- Besoins d'informations pertinentes sur leurs activités (financières, comptables, ...).

II. Généralités sur les BD

■ Exemples de BD :

- **Gestion des personnels, des étudiants, des cours et des inscriptions d'une université ou d'une école ;**
- **Système de réservation des places d'avion des compagnies aériennes ;**
- **Gestion des comptes des clients d'une société bancaire.**

■ Quand utiliser une BD ?

□ Les données à gérer sont de natures diverses :

- Exemples BD d'une Grande Ecole : étudiants, cours, enseignants, salle...

□ Les données possèdent de nombreux liens entre elles

- Exemple : un étudiant suit un cours, un cours est assuré par un enseignant....

II. Généralités sur les BD

□ 2.2. Définition d'un SGBD

- **Système de Gestion de Base de Données**
- **Interface entre l'utilisateur et les données stockées sur le support de mémorisation.**
- **Logiciel qui permet à l'utilisateur d'interagir avec une BD. Il permet d'organiser les données sur les périphériques et fournit les procédures de sélection et de recherche de ces mêmes données.**

■ Plusieurs sortes de SGBD :

- **Hiérarchiques**
- **Réseau**
- **Relationnel**
- **Objet**
- **Objet-Relationnel**

■ Exemple de SGBD :

- **MS ACCESS, ORACLE, SYBASE, DB2, MySQL...**

■ Principe d'une BD

- **Nombreux utilisateurs**
- **Séparation nette entre la BD elle-même et les utilisateurs conduit à la notion d'environnement client/serveur**

II. Généralités sur les BD

□ 2.3. Architecture C/S

■ Client/Serveur

■ Définition

□ Architecture logicielle ouverte (fidèle aux standards) qui fournit des services distants (BD, messagerie, impression, etc.) à des clients interconnectés via réseaux de manière transparente à l'hétérogénéité des ressources informatiques mises en jeu (ordinateurs, réseaux et logiciels de base).

■ Serveur

□ Dispositif informatique offrant des services à des clients :

- Partage de fichiers ;
- Partage de données (BD) ;
- Hébergeur web ...

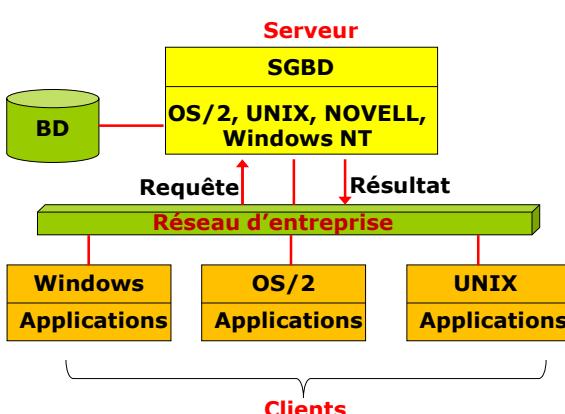
■ Client

□ Dispositif informatique, permettant l'envoi de requêtes à un serveur donné, et la réception des réponses.

□ Interface légère gérant de façon conviviale la communication avec le serveur.

II. Généralités sur les BD

■ Architecture



■ Avantages

□ Ressources centralisées

- Le Serveur gère des ressources communes à tous les utilisateurs.
- Ceci évite les problèmes de redondance.

□ Meilleure sécurité

- Le nombre de points d'entrée permettant l'accès aux données est moins important

□ Administration au niveau serveur

- Les clients ont peu d'importance dans ce modèle, ils ont moins besoin d'être administrés.

□ Réseau évolutif

- Ajout et suppression de clients sans perturber le fonctionnement du réseau.

II. Généralités sur les BD

■ Inconvénients

- **Coût élevé du serveur**
 - Ordinateur puissant
- **Maillon faible**
 - Le serveur est le seul maillon faible du réseau client/serveur,
 - Tout le réseau est architecturé autour de lui.
 - Cependant, le serveur a une grande tolérance aux pannes (notamment grâce au système **RAID**).

II. Généralités sur les BD

□ 2.4. Cycle de vie d'une BD

■ Définition

- **Suite des phases de conception, d'implantation et d'utilisation d'une BD.**

■ Conception d'une BD

- **Le premier problème à résoudre dans la conception d'une BD est de déterminer les informations qu'il conviendra de mettre dans la BD.**
- **Phase d'analyse qui aboutit à déterminer le futur contenu de la BD.**

■ Implantation d'une BD

- **Phase qui consiste à créer la BD et transmettre au SGBD les caractéristiques des informations (structure des tables...) qu'il faut garder pour assurer la gestion de l'entreprise.**
- **On utilise un Langage de Description de Données (LDD).**
- **Après la description des données, suit la saisie des données.**
- **Elle se fait au moyen d'un Langage de Description de Données (LDD) :**
 - Crédit (CREATE), modification (ALTER), et suppression (DROP) des tables...

II. Généralités sur les BD

■ Utilisation d'une BD

- Utilisation de deux langages : LID et LMD.
- LID : Langage d'Interrogation de Données (**SELECT**).
 - Interrogation afin d'obtenir des informations de la BD.
- LMD : Langage de Modification de Données.
 - Mise à jour des données.
 - Ajouter de nouvelles informations (**INSERT**) ;
 - Supprimer des données périmées (**DELETE**) ;
 - Modifier le contenu des données (**UPDATE**)

II. Généralités sur les BD

□ 2.5. Modélisation

- Consiste à créer une représentation simplifiée d'un problème (modèle).
- Comporte deux parties :
 - L'analyse ;
 - La conception.
- Modèle :
 - Représentation simplifiée d'une réalité sur laquelle on veut être renseigné (plans, carte, schémas électroniques).
 - S'exprime avec un ensemble de concepts, dotés de règles d'utilisation et de représentation (souvent graphiques).

■ Analyse

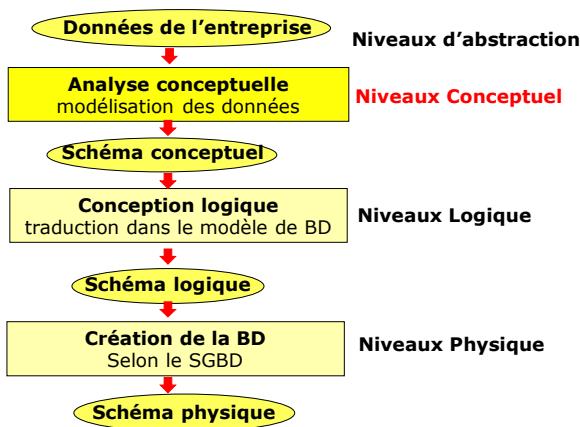
- Comprendre et modéliser le fonctionnement d'un domaine de gestion d'une organisation ;
- Concevoir la solution informatique adéquate.

■ Conception

- Consiste à définir une solution informatique du problème à résoudre.
- Taches à réaliser :
 - Structuration des données ;
 - Organisation des traitements ;
 - Définition des postes de travail ;
 - Choix techniques : matériels, SGBD...

II. Généralités sur les BD

■ Démarche de conception



□ Niveau conceptuel : représente le contenu de la BD en termes conceptuels, indépendamment de toute considération informatique.

▪ Etudié dans le chapitre suivant.

□ Niveau logique relationnelle : résulte de la traduction du schéma conceptuel en un schéma propre à un type de BD.

□ Niveau physique : est utilisé pour décrire les méthodes d'organisation et d'accès aux données de la BD.

III. Le modèle Entité Association

□ 3.1. Modèle E/A

- Anglais : Entity-Relationship
- Représentation statique des données du Système d'Information (SI).
 - Traitement = représentation dynamique des données
- Permet de représenter les données fondamentales de l'entreprise par des règles de gestion d'une manière compréhensible par les utilisateurs.

- SI : Ensemble des moyens humains, matériels et des méthodes de traitement des différentes formes d'information d'une organisation.

III. Le modèle Entité Association

■ Modèle de donnée :

- Ensemble de concepts et de règles de composition de ces concepts permettant de décrire l'information structurée d'un Système d'Information.
- Deux grandes classes de modèle de données :
 - Modèles liés à un SGBD : Modèles hiérarchique, réseau, relationnel...
 - Modèles non liés à un SGBD : Modèles Zo, **Entité/Association**
 -

III. Le modèle Entité Association

■ Crée en 1976 grâce aux travaux menés parallèlement aux Etats Unis et en France.

- Sa paternité est attribuée à Peter Chen.
- Nommé aussi entité/relation, objet/relation

■ A donné naissance à la méthode d'analyse MERISE.

- Méthode d'Etude et de Réalisation Informatique par Sous-Ensemble.
- Méthode de conception et de développement de systèmes d'information.

■ Raisons de son succès :

- Approche naturelle et facile à comprendre ;
- Obtention directe, sans manipulation de données, de résultats de synthèse ;
- Représentation graphique et claire des résultats.

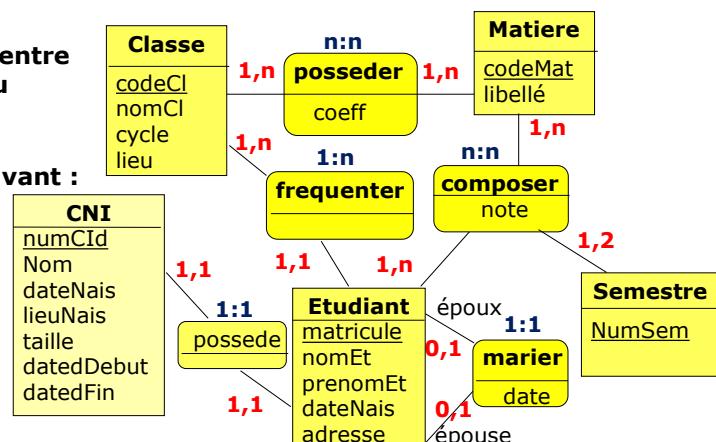
■ Remarque :

- Appelé Modèle Conceptuel de Donnée (**MCD**) en MERISE.

III. Le modèle Entité Association

■ Exemple de Schéma Entité/Association

- Constitué d'**entités** reliées entre elle par des **associations ou relations**.
- Exemple le SI d'une école modélisé par le schéma suivant :



III. Le modèle Entité Association

□ 4.2. Notion d'entité

■ Définition

- Représentation d'un objet du monde réel (concept concret ou abstrait), perçu par le concepteur comme ayant une existence.
- Une entité a une existence propre
 - Indépendamment du fait qu'elle puisse être liée à d'autres entités de la base.

■ Exemples :

- **Etudiant**
- **Classe**

■ Occurrence d'une entité

- L'entité étant considérée comme un ensemble, une occurrence d'une entité est tout élément individualisé de cet ensemble.
- Exemple :
 - **Toto** est une occurrence de l'entité **Etudiant**.
 - **Info1** est une occurrence de l'entité **Classe**.

III. Le modèle Entité Association

■ Représentation

Nom de l'entité

■ Exemple : entité Etudiant

Etudiant

III. Le modèle Entité Association

□ 4.3. Notion d'attribut

- Ou de propriété
- Information élémentaire que l'on peut rattacher à une entité ou à une association.
- C'est une propriété caractéristique de l'entité.
- Représentation

Entité
attribut ₁
attribut ₂
attribut _n

■ Exemple :

□ Définition en intension

Etudiant
matricule
nom
prenom
dateNais
adresse

■ Occurrence de l'entité Etudiant

□ Définition en extension

matricule	Nom	Prénom	Datenaïs	Adresse
M1	Koné	Fatou	26/04/85	Yakro
M2	Koffi	Yao	13/12/84	Bouaké

III. Le modèle Entité Association

□ 3.4. Identifiant d'une entité

- ou clé d'une entité
 - Attribut ou ensemble d'attributs permettant d'identifier de manière unique les occurrences d'une entité.
 - Deux occurrences d'une entité, ne peuvent pas avoir la même valeur de la clé.

■ Exemple :

- L'attribut **matricule** est la clé de l'entité Etudiant

■ Représentation

- La clé d'une entité est toujours soulignée.

■ Exemple :

Etudiant
matricule
nom
prenom
dateNais
adresse

matricule	Nom	Prénom	Datenaïs	Adresse
M1	Koné	Fatou	26/04/85	Yakro
M2	Koffi	Yao	13/12/84	Bouaké
M1	Frete	Mankeu	02/12/85	Yakro

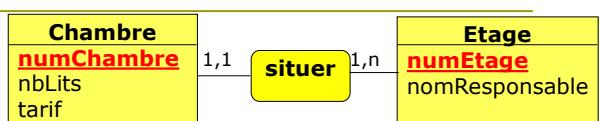
Interdit car doublon dans la clé

- Deux étudiants ne peuvent avoir le même numéro matricule.

III. Le modèle Entité Association

■ Identification relative

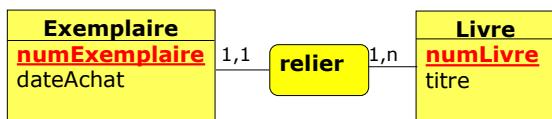
- Traduit le fait qu'une entité "faible" utilise une partie d'une autre entité dite entité "forte" pour s'identifier.
 - Une entité E est dite faible si aucun sous-ensemble de ses attributs ne constitue un identifiant (il n'a pas d'identifiant propre), et qu'un identifiant peut être défini en intégrant un identifiant d'une autre entité E' qui lui est lié par une association binaire de cardinalité 1, 1 pour E.
 - On dit que E dépend de E'.
 - Exemple : chambre dans un hôtel



- Les chambres sont numérotées (1, 2, 3,...) par rapport à l'étage.
 - numChambre seul ne permet donc pas d'identifier la chambre.
 - Pour identifier une chambre, il faut lui adjoindre le numéro de l'étage, ainsi :
 - 11 désignera la chambre 1 situé au 1^{er} étage.
 - 12 désignera la chambre 1 situé au 2^{ème} étage.
 - L'identifiant d'une chambre sera (**numChambre**, **numEtage**).

III. Le modèle Entité Association

□ Exemple 2 : les exemplaires d'un livre



- Les exemplaires sont numérotés (1, 2, 3,...) par rapport au livre.
- Le numéro de l'exemplaire seul ne permet donc pas d'identifier un exemplaire.

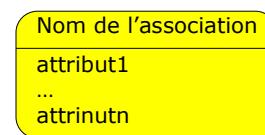
- Pour identifier une chambre, il faut lui adjoindre le numéro du livre, ainsi :
- 11 désignera l'exemplaire 1 du au 1^{er} livre.
- 12 désignera la l'exemplaire 2 du 2^{ème} livre.
- L'identifiant d'un exemplaire sera (**numExemplaire, numLivre**).

III. Le modèle Entité Association

□ 3.5. Notion d'association

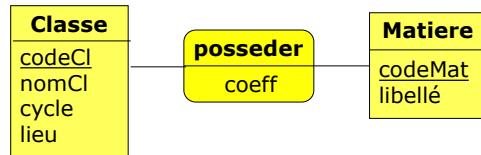
- Traduit dans le SI le fait qu'il existe une relation entre plusieurs entités.
- Dimension de l'association : Nombre d'entité intervenant dans la relation.
 - Association de dimension 2 : association binaire.
 - Association de dimension 3 : association ternaire.
 - Association de dimension n : association n-aire.

■ Représentation



□ Rectangle aux bords arrondis

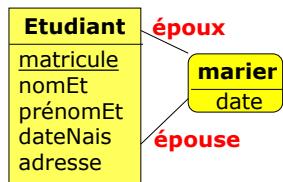
■ Exemple : association binaire



III. Le modèle Entité Association

■ Association réflexive

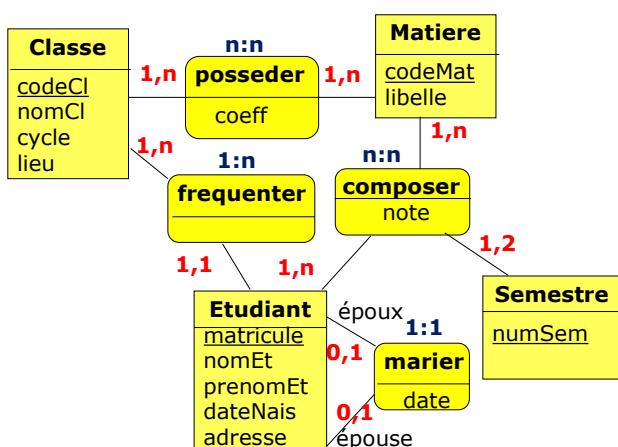
- Lie deux entités de même type.
- Indiquer le rôle de chaque entité pour supprimer les ambiguïtés.
- Exemple :



- époux et épouse sont les rôles de l'association "marier"

III. Le modèle Entité Association

■ Clé d'une association



- Obtenu par la réunion des identifiants des entités qui participent à l'association.
- Un attribut d'association ne peut pas être déclarée comme clé de l'association.

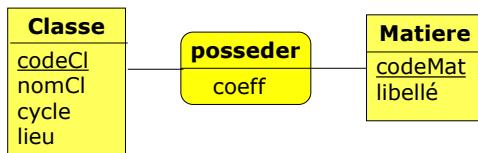
■ Exemples :

Association fréquenter : (matricule, codeCl)
Association posséder : (codeCl, codeMat)
Association Composer : (matricule, codeMat, numSem)
Association marier : (matriculeEpoux, matriculeEpouse)

III. Le modèle Entité Association

■ Occurrence d'une association

- Ses attributs sont les attributs de l'association et les clés des entités participant à l'association.
- Exemple 1 : occurrences de l'association posséder



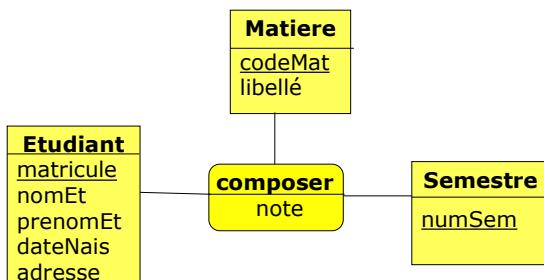
■ Occurrences de Posséder

codeCl	codeMat	coeff
C1	M1	2
C2	M1	3
C3	M1	1
C3	M2	2
C2	M1	5

Interdit car doublon dans la clé

III. Le modèle Entité Association

■ Exemple 2 : occurrences de l'association composer



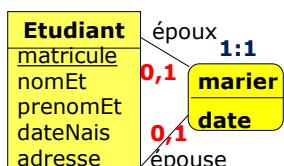
■ Occurrences de Composer

matricule	codeMat	numSem	Note
M007	01	1	15,50
M007	02	1	19
M007	01	2	14
M059	01	1	16
M059	01	1	15

Interdit car doublon dans la clé

III. Le modèle Entité Association

- Exemple 3 : occurrences de l'association marier



- Occurrences de Composer

matricoleEpoux	matricoleEpouse	date
M01	M02	2010
M03	M05	2010
M80	M04	2012
M03	M20	2016
M03	M05	2012

(M03, M20 : Interdit à cause de la cardinalité max 1)

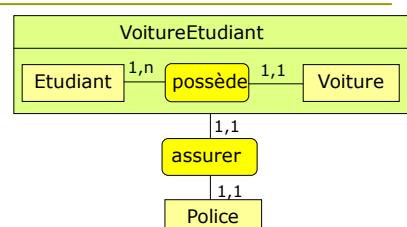
(M03, M05) : Interdit car doublon dans la clé

III. Le modèle Entité Association

- Association d'association

- Ou pseudo-entité
- Il s'agit d'une association qui est vue comme une entité et qui doit être reliée à une autre association.
- Permet de créer des entités plus complexes lorsqu'une entité est liée à un groupe d'entités.
- Exemple 1 :

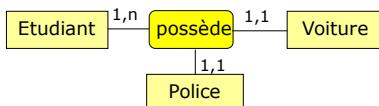
- les polices d'assurance sur le campus ne concernent que les voitures des étudiants.
- On peut exprimer ce fait à l'aide d'une pseudo-entité qui créera une entité composée VoitureEtudiant.



- La pseudo-entité VoitureEtudiant n'a de réalité pour l'association est assurée que parce que c'est le couple d'entité (Etudiant, Voiture) qui est concerné par l'assurance.
- En effet une police est nominale, elle ne concerne un véhicule et un propriétaire, en l'occurrence l'étudiant.

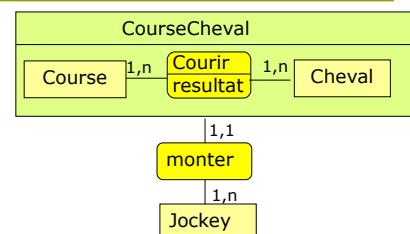
III. Le modèle Entité Association

- On peut transformer cette association en association ternaire.

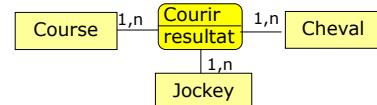


Exemple 2 : course de chevaux

- Un jockey peut monter différents chevaux dans différentes courses.
- Cependant, pour une course donnée, un jockey n'utilise qu'un seul cheval.



- Association ternaire



III. Le modèle Entité Association

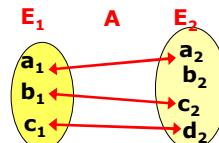
3.6. Type d'association

- Nombre maximum d'occurrence mise en jeu de part et d'autre d'une association.**
- Trois types d'association :**
 - Un à un **1:1**
 - Un à plusieurs : **1:n**
 - Plusieurs à plusieurs : **n:n**

III. Le modèle Entité Association

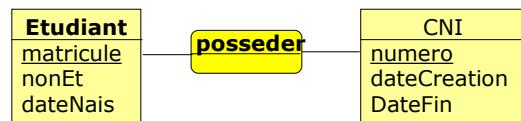
■ Association un à un

- Soient E_1 et E_2 deux entités liées par l'association A .



- A est une association un à un si à une occurrence de l'entité E_1 peut correspondre par l'association A au plus une occurrence de l'entité E_2 .**
- Réciproquement, à une occurrence de l'entité E_2 ne peut correspondre qu'au plus une occurrence de l'entité E_1 .**

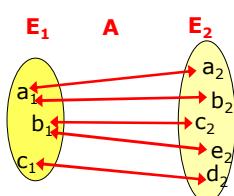
■ Exemple :



- "**Posséder**" une association de type un à un.
- En effet, un Etudiant possède une et une seule CNI et réciproquement à une CNI n'est associée qu'un seul Etudiant.

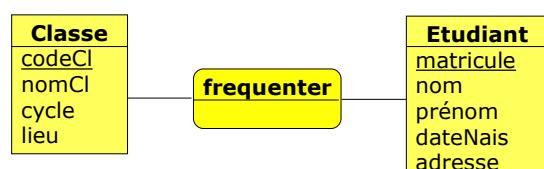
III. Le modèle Entité Association

■ Association un à plusieurs



- A est une association un à plusieurs si à une occurrence de l'entité E_1 , l'association A peut faire correspondre plusieurs occurrences de l'entité E_2 mais qu'à une occurrence de E_2 correspond au plus une occurrence de E_1 .**

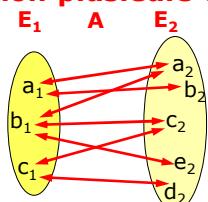
■ Exemple :



- "**fréquenter**" est une association de type 1 à plusieurs.
- En effet, dans une classe peuvent fréquenter plusieurs étudiants et un étudiant ne fréquente que dans une seule classe.

III. Le modèle Entité Association

■ Association plusieurs à plusieurs



■ A est une association plusieurs à plusieurs si à une occurrence de l'entité E₁ peuvent être associées, par l'association A, plusieurs occurrences de l'entité E₂, et réciproquement, à une occurrence de l'entité E₂ peuvent être associées, par l'association A plusieurs occurrences de l'entité E₁.

■ Exemple :



- "Posséder" est une association de type plusieurs à plusieurs.
- En effet, une classe peut posséder plusieurs matières et une matière peut être enseignée dans plusieurs classes.

III. Le modèle Entité Association

■ Représentation



■ Exemple :



III. Le modèle Entité Association

□ 3.7. Cardinalité

■ Cardinalité d'un couple entité/association

- Précise le nombre de fois minimum et maximum qu'une occurrence d'entité participe à une association.
- Caractérisée par une borne minimale et une borne maximale.

■ Cardinalité minimale :

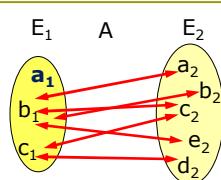
- Nombre minimum de fois qu'une occurrence d'une entité participe aux occurrences de l'association.

■ Cardinalité maximale :

- Nombre maximum de fois qu'une occurrence d'une entité participe aux occurrences de l'association.

III. Le modèle Entité Association

■ Exemples :



□ Cardinalité min

- Dans la figure ci-dessus, aucun trait ne lie l'occurrence a_1 de E_1 alors que plusieurs traits lient les occurrences b_1 et c_1 .
- Cela signifie que pour une occurrence donnée de E_1 (exemple a_1) on peut n'avoir aucune association avec les occurrences de E_2 i.e. au minimum zéro trait liant l'occurrence.

□ Cardinalité max

- Par contre il existe une autre occurrence de E_1 (ex b_1 ou c_1) pour laquelle on peut avoir plusieurs occurrences de E_2 .
- On dit que la cardinalité de E_1 , muni de l'association A est $(0, n)$; de même la cardinalité de E_2 , muni de l'association A est $(1, n)$.

III. Le modèle Entité Association

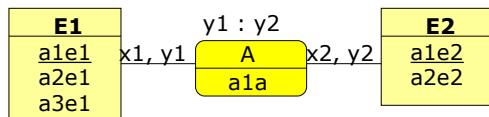
■ Notation :

- Les cardinalités les plus courantes sont :
 - 0, 1 : aucun ou un seul
 - 1, 1 : un et un seul
 - 0, n : aucun ou plusieurs
 - 1, n : au moins un ou plusieurs

■ Représentation



■ Représentation générale

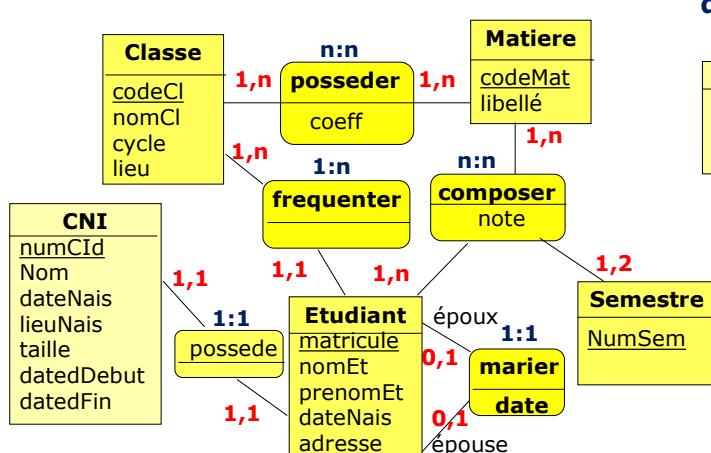


□ Exemple :

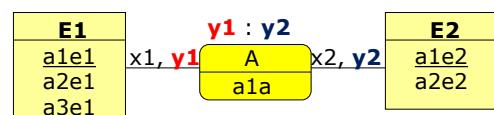


III. Le modèle Entité Association

■ Exemple récapitulatif :



■ Lien entre cardinalité et type d'association



III. Le modèle Entité Association

□ 3.8. Héritage

- Association particulière
- Lien orienté, d'une entité spécialisée (ou spécifique ou fille) vers une autre entité générique (ou père).
- Il permet d'exprimer les caractéristiques communes à plusieurs entités formant une même famille.

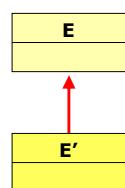
- Il modélise le lien "est-un" ou "is-a" entre deux entités E et E'.

- E est appelée entité générique (ou entité sur-type)
 - E' est appelée entité spécialisée (ou entité sous-type).
- Sémantique du lien :
 - A toute occurrence de l'entité spécifique correspond une occurrence de l'entité générique qui décrit le même objet du monde réel ;
 - Inversement, à toute occurrence de l'entité générique correspond zéro ou une occurrence de l'entité spécifique.

III. Le modèle Entité Association

- Représentation

- une flèche allant de E' vers E.



- L'entité E est plus générale que l'entité E'.

- Principale propriété de la généralisation.

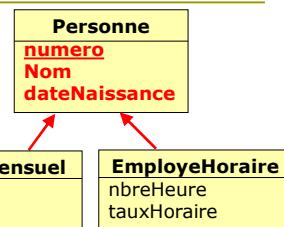
- Toute entité E' ayant avec une autre entité E une relation de généralisation dans le sens E' est un E hérite des attributs de E.
 - Les attributs adjoints à l'entité E' sont appelés attributs propres et ne peuvent être hérités par l'entité E.

III. Le modèle Entité Association

- Exemple :



- Attribut communs : **nom** et **dateNaissance**

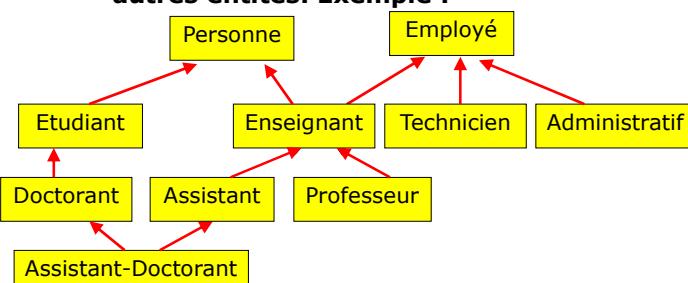


- Les entités **EmployeMensuel** et **EmployeHoraire** vont hériter des attributs **numero**, **nom** et **dateNaissance** de l'entité **Personne**.
- Par contre l'entité **Personne** ne peut hériter de l'attribut **salaire** car toutes les personnes ne sont pas des Employés salariés.

III. Le modèle Entité Association

- Héritage multiple

- Une entité hérite de plusieurs autres entités. Exemple :



- L'entité **Assistant-Doctorant**, regroupe la population des personnes qui sont à la fois **Assistant** et **Doctorant**.

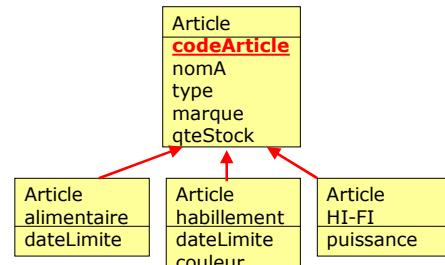
- Problème de la généralisation multiple

- pose des problèmes liés à l'héritage.
- Eviter d'hériter deux fois d'un ancêtre commun.
- L'entité **Assistant-Doctorant** doit hériter une seule fois de l'entité **Personne** :
 - Soit via Doctorant et Etudiant,
 - Soit via Assistant-Enseignant-Employé.

III. Le modèle Entité Association

■ Identifiant d'une sous-classe

- Soit E' une entité sous-classe de l'entité E, alors tout identifiant de E est aussi identifiant de E'.
- E' n'a pas nécessairement d'identifiant qui lui soit propre.
- Dans l'exemple de hypermarché, Article, Article alimentaire, Article habillement et Article HI-FI ont tous les trois pour identifiant celui de Article (**codeArticle**).



III. Le modèle Entité Association

□ Exercice 1 : QCM

- **Q1 : Une entité est caractérisée par :**
 - a) Nom de l'entité, attributs, valeurs.
 - b) Nom de l'entité, identifiants, valeurs.
 - c) Nom de l'entité, clés, valeurs.
- **Q2 : Une entité peut avoir plusieurs identifiants.**
 - a) Vrai
 - b) Faux

■ Q3 : C'est quoi une occurrence d'entité ?

- a) Chaque élément de l'entité.
 - b) Le premier élément de l'entité.
 - c) Le dernier élément de l'entité.
- ### ■ Q4 : La cardinalité d'un couple Entité/Association :
- a) Est représentée par plusieurs valeurs.
 - b) Précise le nombre de fois minimum et maximum qu'une occurrence d'entité participe à une association.
 - c) Le nombre d'occurrence d'une entité ou d'une association.

III. Le modèle Entité Association

- **Q5 : le type d'association peut être déduit des cardinalités.**
 - a) Vrai
 - b) Faux
- **Q6 : Dans un lien d'héritage, les entités sous-type n'ont pas d'identifiants propres.**
 - a) Vrai
 - b) Faux
- **Q7 : Dans la conception d'une BD, au niveau conceptuel, on :**
 - a) Recense les règles d'organisation.
 - b) Définit le type de base données.
 - c) Recense les règles de gestion.
 - d) Défini l'architecture réseau
- **Q8 : Quelle est l'affirmation fausse ?**
 - a) Une propriété peut se retrouver dans plusieurs endroits du modèle.
 - b) Une relation peut relier plus de deux objets.
 - c) Une ou plusieurs propriétés permettent d'identifier de manière unique un objet.
 - d) La dimension d'une relation est le nombre d'objets rentrant dans la liaison.

III. Le modèle Entité Association

- **Q9 : Un identifiant :**
 - a) Un attribut qui n'accepte pas de doublon.
 - b) Permet d'identifier de façon unique une occurrence d'une entité.
 - c) Première occurrence d'une entité.
- **Q10 : La modélisation :**
 - a) Est une étape de réflexion avant la conception.
 - b) Le modèle Entité/Association est un schéma qui vise à rendre compte des relations existant entre divers éléments d'un système.
 - c) Permet d'éviter les anomalies d'insertion et de modifications
 - d) Une entité possède toujours au moins un attribut.

III. Le modèle Entité Association

■ Q11 : Une association :

- a) Est un lien sémantique non orienté ayant au moins 2 brins et 2 couples de cardinalités.
- b) Elle s'effectue toujours entre deux entités.
- c) Le degré d'une association est le nombre de lignes.
- d) La dimension d'une relation est le nombre de colonnes.
- e) Elle ne peut avoir de propriétés.

■ Q12 : Dans la conception d'une BD relationnelle, on ne doit pas :

- a) dégager les entités et leur identifiant.
- b) dégager les relations et leur identifiant.
- c) déterminer les cardinalités des relations.
- d) attacher les propriétés aux relations et aux objets.

III. Le modèle Entité Association

□ Exercice 2 :

■ Coopérative de consommateurs

Une coopérative de consommateurs reçoit des commandes de ses adhérents et s'approvisionne auprès de ses fournisseurs pour satisfaire des commandes. Par exemple, Koffi Yao a commandé 1000 kg d'ignames 5 sacs de gombos à la coopérative. Celle-ci choisi d'acheter les ignames à madame Koné à Korhogo au tarif 1000 le kg et des ignames à monsieur Koffi à Bouaké à 800 le kg et les gombos à madame Tra Lou à Zuenoula.

Questions :

1. Déterminer les entités
2. Déterminer les associations
3. Faire le schéma Entité/Association
4. Donner des occurrences des entités et des associations.
5. Envisager un point de vue diachronique (historique) de la réalité de la coopérative. On doit pouvoir acheter le même produit plusieurs fois à des périodes différentes.

III. Le modèle Entité Association

□ Exercice 3 : Bibliothèque

Une bibliothèque universitaire souhaite automatiser sa gestion. Cette bibliothèque est gérée par un gestionnaire chargé des inscriptions et des relances des lecteurs quand ceux-ci n'ont pas rendu leurs ouvrages au-delà du délai autorisé. Les bibliothécaires sont chargés de gérer les emprunts et la restitution des ouvrages ainsi que l'acquisition de nouveaux ouvrages. Il existe trois catégories d'abonné. Tout d'abord les étudiants qui doivent seulement s'acquitter d'une somme forfaitaire pour une année afin d'avoir droit à tous les services de la bibliothèque. L'accès à la bibliothèque est libre pour tous les enseignants. Enfin, il est possible d'autoriser des étudiants d'une autre université à s'inscrire exceptionnellement comme abonné moyennant le versement d'une cotisation.

Le nombre d'abonnés externes est limité chaque année à environ 10% des inscrits. Un nouveau catalogue général des ouvrages doit être mis en place. Les ouvrages, souvent acquis en plusieurs exemplaires, sont rangés dans des rayons de la bibliothèque. Chaque exemplaire est repéré par une référence gérée dans le catalogue et le code du rayon où il est rangé. Chaque abonné ne peut emprunter plus de trois ouvrages. Le délai d'emprunt d'un ouvrage est de trois semaines, il peut cependant être prolongé exceptionnellement à cinq semaines.

Questions :

1. Déterminer les entités
2. Déterminer les associations
3. Faire le schéma Entité/Association

III. Le modèle Entité Association

□ Exercice 4 : Gestion des formations

Il s'agit d'établir le schéma conceptuel d'une base de données pour la gestion des formations d'un institut privé. Un cours est caractérisé par un numéro de cours (NOCOURS), un libellé (LIBELLE), une durée en heures (DUREE) et un type (TYPE). Un cours peut faire l'objet dans l'année de plusieurs sessions identiques. Une session est caractérisée par un numéro (NOSES), une date de début (DATE) et un prix (PRIX). Une session est le plus souvent assurée par plusieurs animateurs et est placée sous la responsabilité d'un animateur principal. Un animateur peut intervenir dans plusieurs sessions au cours de l'année. On désire mémoriser le nombre d'heures (NBH) effectué par un animateur pour chaque session. Un animateur est caractérisé par un numéro (NOANI), un nom (NOMA) et une adresse (ADRA).

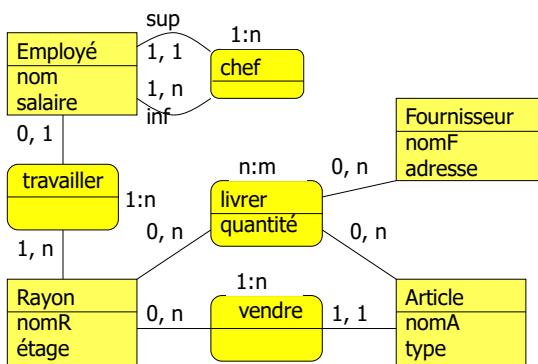
Chaque session est suivie par un certain nombre de participants. Un participant est une personne indépendante ou un employé d'une entreprise cliente. Un participant est caractérisé par un numéro (NOPAR), un nom (NOMP) et une adresse (ADRP). Dans le cas d'un employé, on enregistre le nom (NOMEN) et l'adresse de l'entreprise (ADREN). On désire pouvoir gérer d'une manière séparée (pour la facturation notamment) les personnes indépendantes d'une part, et les employés d'autre part. Si nécessaire, on fera les hypothèses sémantiques complémentaires qui pourraient s'imposer.

Questions :

1. Déterminer les entités
2. Déterminer les associations
3. Faire le schéma Entité/Association

III. Le modèle Entité Association

□ Exercice 5 : Hypermarché



On veut séparer les articles (d'attributs n°Code, nomA, marque, qté en stock) en plusieurs classes : article alimentation (d'attribut date limite de vente), article habillement (d'attribut taille et couleur), HI-FI (d'attribut puissance), hygiène... On introduit également une entité Service après vente pour la réparation des articles HI-FI.

Questions :

- Compléter le schéma E/A de l'hypermarché avec cette nouvelle réalité qui pourraient s'imposer.

IV. Modèle Relationnel

□ 4.1. Présentation

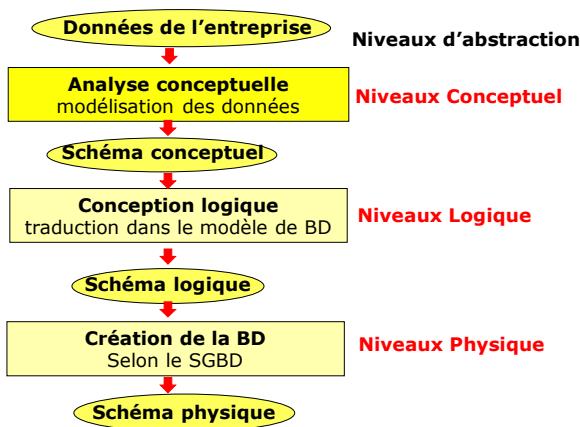
- Introduit par Edgar Frank Codd en 1970.
- Fondé sur la notion mathématique de relation.
- Données organisées sous forme de tables à deux dimensions encore appelées relations.
 - Chaque ligne est appelée n-uplet ou tuple.
- Exemple : relation Produit

numProd	designation	prix	qteStock
1	CD	400	12
2	DVD	500	20

- Données manipulées par des opérateurs de l'algèbre relationnelle (union, intersection, ...) et SQL.
- Etat cohérent de la BD assuré par un ensemble de CI.
- Théorie de la normalisation
 - Permet d'éliminer les incohérences lors de la conception d'une BD.

IV. Modèle Relationnel

■ Démarche de conception



- Niveau conceptuel : représente le contenu de la BD en termes conceptuels, indépendamment de toute considération informatique.
- Niveau logique relationnelle : résulte de la traduction du schéma conceptuel en un schéma propre à un type de BD.
 - Etudié dans ce cours
- Niveau physique : est utilisé pour décrire les méthodes d'organisation et d'accès aux données de la BD.

IV. Modèle Relationnel

□ 4.2. Quelques définitions

■ Notion de Domaine :

- Ensemble de valeurs
- Exemples :
 - Entier
 - Réel
 - Chaîne de caractères
 - Date

■ Produit cartésien :

- Le produit cartésien d'un ensemble de domaine D_1, D_2, \dots, D_n , noté $D_1 \times D_2 \times \dots \times D_n$ est l'ensemble de n-uplets ou tuples (d_1, d_2, \dots, d_n) tels que $d_i \in D_i$ avec $i = 1, 2, \dots, n$.
- Exemple :
 - $D_1 = \{BD, Math, IA\}$ card(D_1)=3
 - $D_2 = \{3, 2\}$ card(D_2)=2
 - $D_1 \times D_2 : \text{card}(D_1 \times D_2) = 3 \times 2$

BD	3
BD	2
MATH	3
MATH	2
IA	3
IA	2

IV. Modèle Relationnel

■ Notion de Relation

- La relation sur les attributs A_1, A_2, \dots, A_n de domaine respectif D_1, D_2, \dots, D_n est tout sous-ensemble de produit cartésien de D_1, D_2, \dots, D_n .
- C'est donc tout ensemble de n-uplets (x_1, x_2, \dots, x_n) , avec x_i un élément de D_i ($1 \leq i \leq n$).
- Exemples :

Relation 1 : {(BD, 3), (Math, 2)}
Relation 2 : {(BD, 3), (IA, 3)}

■ Notations :

- Relation R définie sur les domaines D_1, D_2, \dots, D_n :
 - $R(D_1, D_2, \dots, D_n)$
- Prise en compte du concept d'attribut :
 - $R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$
- Notation utilisée en pratique :
 - $R(A_1, A_2, \dots, A_n)$ avec $A_i =$ attributs de la relation

IV. Modèle Relationnel

■ Exemple : relation Etudiant

- Définition en intention

Etudiant(numEt, nomEt, dateNais, adresse, nomCl)

- Définition en extension

Attributs					
Etudiant	numEt	nomEt	dateNais	adresse	nomCl
	214	Toto	26/10/72	Abidjan	INFO1
	166	Froto	11/02/80	Abidjan	INFO2
	108	Mankou	15/04/79	Bouaké	INFO1

■ Remarques :

- Un tableau ne contient pas deux lignes identiques
- L'ordre des lignes n'a pas d'importance
- L'ordre des colonnes n'a pas d'importance
- Chaque case du tableau ne contient qu'une seule valeur
- nombre de tuples d'une relation = cardinalité de la relation
- nombre d'attributs d'une relation = degré de la relation

IV. Modèle Relationnel

□ 4.3. Clé de relation

- Tout sous-ensemble **minimal** d'attributs dont les valeurs permettent de distinguer les tuples les uns des autres.
- **Clé candidate** : Toute clé potentielle.
- **Clé primaire d'une relation** : une des clé candidates.
- **Exemple :**
 - Dans la relation Etudiant, on peut avoir comme clé numEt (numéro de l'étudiant)

■ Clé étrangère

- ou identifiant externe
- Attribut permettant de référencer les tuples d'une autre relation ou parfois la même.
- Leur valeur est nécessairement égale à celle de la clé d'un tuple existant dans l'autre relation.

□ Exemple :

- nomCl dans la relation Etudiant

Classe

nomCl	salle
INFO1	369
INFO2	360
INFO3	312

IV. Modèle Relationnel

□ 4.4. Contrainte d'intégrité

- Prédicat permettant de vérifier la validité des opérations de mise à jour afin que la BD soit toujours cohérente.
- **Exemples :**
 - Contrainte d'intégrité de domaine
 - Contrainte d'intégrité de relation ou de clé primaire
 - Contrainte d'intégrité de référence

IV. Modèle Relationnel

□ 4.5. Schéma de relation

- Le doublet U des attributs et de l'ensemble F des contraintes d'intégrité d'une relation.
- $R = \langle U, F \rangle$
- Exemple :
 - Etudiant = $\langle U_1, F_1 \rangle$
 - avec $U_1 = \{\text{numEt}, \text{nom}, \text{dateNaiss}, \text{adresse}\}$
 - $F_1 = \{\text{numEt est un entier positif}, \text{dateNaiss} < \text{date du jour}\}$
 - Tout ensemble de tuples construit sur U_1 et vérifiant F_1 est une occurrence du schéma de relation.

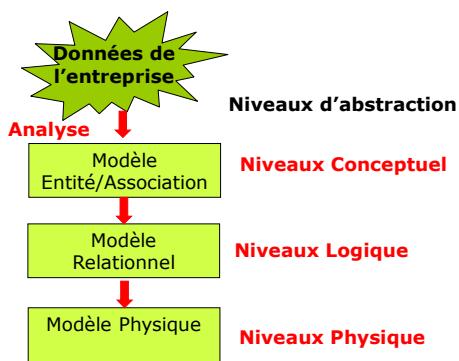
■ Schéma de la BD ou schéma relationnel :

- L'ensemble des schémas de relation de la BD.
- Exemple :
 - Etudiant = $\langle U_1, F_1 \rangle$
 - Classe = $\langle U_2, F_2 \rangle$

IV. Modèle Relationnel

□ 4.6. Conception d'un schéma relationnel

■ Démarque de conception

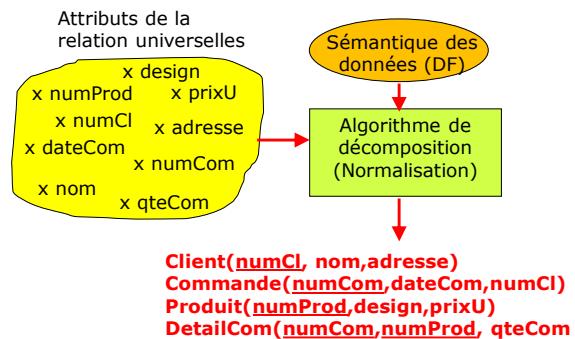


- Niveau conceptuel : représente le contenu de la BD en termes conceptuels, indépendamment de toute considération informatique.
- Niveau logique relationnelle : résulte de la traduction du schéma conceptuel en un schéma propre à un type de BD.
- Niveau physique : est utilisé pour décrire les méthodes d'organisation et d'accès aux données de la BD.

IV. Modèle Relationnel

■ Principe

- Une mauvaise conception peut engendrer des problèmes dans son exploitation (redondance, MAJ)
- La théorie de normalisation est basée sur les DF.
- Les DF traduisent des contraintes sur les données



IV. Modèle Relationnel

□ 4.7. DF

- Dépendance fonctionnelle : DF
 - Soit une relation $R(A, B, C)$ où A , B et C représentent des attributs ou groupes d'attributs. B est dit fonctionnellement dépendant de A si étant donné deux tuples $\langle a_1, b_1, c_1 \rangle$ et $\langle a_2, b_2, c_2 \rangle$ de R , $a_1 = a_2 \Rightarrow b_1 = b_2$.
 - Plus simplement, B dépend fonctionnellement de A , si étant donné une valeur de A il lui correspond une valeur unique de B .
 - La DF est notée : $A \rightarrow B$ et se lit : A détermine B ou B est fonctionnellement dépendant de A .

□ Exemple :

`CommandeProduit(numProd, design, qteCom, prixU, numCom, dateCom, numCl, nom, adresse)`

```
F = {numProd → design ;
     numProd → prixU ;
     numCl → nom ;
     numCl → adresse ;
     numCom → dateCom ;
     numCom → numCl ;
     numCom, numProd → qteCom}
```

□ Contre exemples :

- $numCl \rightarrow numProd$
- $numCom \rightarrow qteCom$

IV. Modèle Relationnel

Propriétés des DF

- Règles d'Amstrong

Réflexivité :

$Y \subseteq X \Rightarrow X \rightarrow Y$ (dépendance triviale)

Augmentation :

$X \rightarrow Y \Rightarrow X, Z \rightarrow Y, Z$

Transitivité :

$X \rightarrow Y$ et $Y \rightarrow Z \Rightarrow X \rightarrow Z$

Union ou Additivité :

$X \rightarrow Y$ et $X \rightarrow Z \Rightarrow X \rightarrow Y, Z$

Pseudo-transitivité :

$X \rightarrow Y$ et $Y, W \rightarrow Z \Rightarrow X, W \rightarrow Z$

Décomposition :

$X \rightarrow Y$ et $Z \subseteq Y \Rightarrow X \rightarrow Z$

Dépendance Fonctionnelle Élémentaire (DFE)

- Une DF $X \rightarrow Y$ est élémentaire s'il n'existe pas de $X' \subset X$ tel que $X' \rightarrow Y$**
- Exemple :

$\text{numCl} \rightarrow \text{nom}$;
 $\text{numCom}, \text{numProd} \rightarrow \text{qteCom}$

- Contre exemple :

$\text{numCl}, \text{numCom} \rightarrow \text{nom}$
 $\text{car numCl} \rightarrow \text{nom}$

IV. Modèle Relationnel

Dépendance Fonctionnelle Directe (DFD)

- Une DF $X \rightarrow Y$, dans une relation R est directe s'il n'existe pas d'attribut Z de R tel que $X \rightarrow Z$ et $Z \rightarrow Y$**
- La DF $X \rightarrow Y$ n'est pas obtenue par transitivité
- Exemples :

$\text{numCl} \rightarrow \text{nom}$;
 $\text{numCom}, \text{numProd} \rightarrow \text{qteCom}$

- Contre exemple :

$\text{numCom} \rightarrow \text{nom}$
 $\text{car numCom} \rightarrow \text{numCl}$ et $\text{numCl} \rightarrow \text{nom}$

Fermeture transitive d'un ensemble F de DFE

- Ensemble F^+ obtenu en ajoutant à F toutes les DF élémentaires déduites par transitivité.
- $F^+ = F \cup \{\text{DF transitives}\}$

Couverture minimale

- Ensemble MIN(F) de DF élémentaires associé à un ensemble d'attributs**
- vérifie les propriétés suivantes :
 - aucune dépendance de F n'est redondante ; c'est-à-dire, pour toute DF F' de F , $F - \{F'\}$ n'est pas équivalent à F
 - toute DF élémentaire des attributs est dans F^+ de F .
 - En d'autres termes, il n'existe pas $F' \subset \text{MIN}(F)$ / $(F')^+ = F^+$

IV. Modèle Relationnel

□ 4.8. Normalisation

■ Définition

- Transformation d'une relation posant des problèmes lors des opérations de MAJ en plusieurs relations n'ayant pas ces problèmes.
- But : Minimiser les redondances car source d'incohérence de la BD.
- Classification des relations :
 - En fonction de leurs propriétés vis-à-vis des DF

- 5 formes normales : 1FN, 2FN, 3FN, BCNF, 4FN, 5FN
- Degré de normalité d'un schéma
- Plus il est élevé, plus les redondances et les anomalies de MAJ sont réduites.

IV. Modèle Relationnel

■ Première FN (1FN)

- Une relation est normalisée ou en 1FN si tous les attributs qui la composent sont de type élémentaire ou atomique, i.e. non décomposable.
- La notion d'agrégat ou de groupe répétitif (ensemble de valeurs), n'existe pas dans le modèle relationnel.
- Types interdits : structures, ensembles, listes, tableaux...
- Remarques :
 - Toute relation dans une BDR est en 1FN

■ Normalisation en 1FN :

- Eclatement de la relation : on remplace l'attribut composé par les attributs simples correspondants
- Eclatement du groupe répétitif : on réplique les tuples autant de fois qu'il y a de valeur pour un attribut donné

IV. Modèle Relationnel

- Exemple : attribut de type structure (**adresse**)

Personne

id	nom	adresse		
		n°	rue	ville

- Mise à plat de la structure

Personne

id	nom	n°	rue	ville	CP
-----------	------------	-----------	------------	--------------	-----------

- Création de nouvelles relations

Personne

Personne	Adresse	Habite
id	Aid	id

- Exercice : soient les deux relations suivantes :

Avion(avnom, capacité)

avnom	capacité
CARV	100
B707	150
AIRB	250
AIRB	350

Vol(numvol, Avion)

numvol	Avion
100	(CARV, 100)
101	(B707, 150)
102	(B707, 150)
103	(AIRB, 250)
104	(AIRB, 350)

- Normaliser la relation Vol

numvol	avnom
100	CARV
101	B707
102	B707
103	AIRB
104	AIRB

IV. Modèle Relationnel

■ Deuxième FN (2FN)

- Permet d'éliminer certaines redondances dues au fait que certains attributs non-clés sont déterminés par une partie de la clé.
- Une relation est en 2FN si et seulement si :
 - elle est en 1FN
 - tout attribut n'appartenant à aucune clé ne doit pas dépendre d'une partie de cette clé (dépend de la clé que par une DFE)

■ Remarque :

- Si la relation est en 1FN et la clé est unique et mono-attributs alors la relation est en 2FN.

■ Exemple 1 :

Etudiants(numEt, nom, codeCl, libellé), avec
 $F = \{ \text{numEt} \rightarrow \text{nom}, \text{codeCl}, \text{libellé} ; \text{codeCl} \rightarrow \text{libellé} \}$

- La relation Etudiant est en 2FN car la clé est mono attribut

IV. Modèle Relationnel

■ Exemple 2 :

Fournisseur(nom, adresse, article, prix),
avec
 $F = \{ \text{nom} \rightarrow \text{adresse} ;$
 $\quad \text{nom, article} \rightarrow \text{prix}$
}

- La relation Fournisseur n'est pas en 2FN car une partie de la clé (nom) détermine un attribut (adresse) n'appartenant pas à la clé

nom	adresse	article	prix
F1	A1	P1	100
F1	A1	P2	125
F1	A1	P3	500
F2	A2	P1	200

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

77

■ Les anomalies de stockage existent dans Fournisseur pour les opérations de MAJ :

- Anomalie d'insertion** : on ne peut insérer une occurrence du couple (article, prix) si la valeur correspondante au couple (nom, article) constituant la clé primaire n'existe pas (contrainte d'intégrité d'entité)
- On ne peut insérer un nouveau produit si il n'y a pas de fournisseur et vice versa.

77

IV. Modèle Relationnel

- Anomalie de suppression** : si on supprime une valeur de la clé primaire par exemple (F2, P1), on risque de perdre une valeur unique du couple (nom, adresse)
 - par exemple (F2, A2).

nom	adresse	article	prix
F1	A1	P1	100
F1	A1	P2	125
F1	A1	P3	500
F2	A2	P1	200

- Anomalie de modification** : dans la relation Fournisseur il aura généralement une redondance de certaines occurrences du couple (nom, adresse) par exemple (F1, A1). Dès lors, si on veut changer l'adresse d'un fournisseur, on aura :

- Soit un coût élevé de MAJ si la modification est effectuée sur chaque occurrence répliquée ;
- Soit une incohérence de la relation si la modification est effectuée sur une seule occurrence.

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

78

78

IV. Modèle Relationnel

■ Décomposition de la relation

Fournisseur : **Théorème de Heath**

- Soit la relation $R(X, Y, Z)$ où X, Y, Z sont des groupes d'attributs.
Si $X \rightarrow Y$ appartient à F , alors la décomposition de R en $S(X, Y)$ et $T(X, Z)$ est SPI (Sans Perte d'Information).
- Réciproquement, si la décomposition de R en S et T est SPI, alors $X \rightarrow Y$ ou $X \rightarrow Z$ appartient à F^+ .

□ Par application du théorème de Heath, on obtient alors les 2 relations suivantes en 2FN :

- $X = \text{nom}$, $Y = \text{adresse}$,
 $Z = \text{article}, \text{prix}$
- Fournisseur(nom, adresse) avec
 $F = \{\text{nom} \rightarrow \text{adresse}\}$
- Produit(nom, article, prix) avec
 $F = \{\text{nom}, \text{article} \rightarrow \text{prix}\}$

IV. Modèle Relationnel

■ Avantages de la 2FN

- Permet de limiter les redondances.
- Considérons les deux schémas ci-dessous :
 - 100 Fournisseurs et 30 produits différents ; un produit est livré que par 2 fournisseurs maximum et chaque attribut est codé sur 1 octet.

□ Relation en 2FN nécessite moins d'octet pour le stockage des

Schémas	FN	Nombre d'octets
Fournisseurs(<u>nom</u> , <u>adresse</u> , <u>article</u> , <u>prix</u>)	1FN	$100 \times 4 \times 2 = 800$
Fournisseur(<u>nom</u> , <u>adresse</u>) Produit(<u>nom</u> , <u>article</u> , <u>prix</u>)	2FN	$100 \times 2 = 200$ $30 \times 3 \times 2 = 180$ $200 + 180 = 380$

IV. Modèle Relationnel

■ Exercice 1 :

- Soit la relation $R(A, B, C, D, E)$ et $F=\{ B \rightarrow C ; A, B \rightarrow D, E \}$.
- Montrer que R n'est pas en 2FN et décomposé la.

■ Exercice 2 :

- Soit la relation $\text{Vol}(\text{numVol}, \text{jour}, \text{numPl}, \text{numAv}, \text{typeVol})$ et $F = \{\text{numVol} \rightarrow \text{typeVol} ; \text{numVol}, \text{jour} \rightarrow \text{numPl}, \text{numAv} \}$.
- Montrer que R n'est pas en 2FN et décomposé la.

■ Exercice 3 :

- On considère la relation $R(A, B, C, D, E, F)$ sur laquelle sont définies les DF suivantes : $F=\{A, B \rightarrow C ; D \rightarrow C ; D \rightarrow E ; C, E \rightarrow F ; E \rightarrow A\}$.

- Compléter les cases vides du tableau suivant :

A	B	C	D	E	F
	1		110		54
x	2	j	100	n	52
	1	i	110	m	
	2		100		52
	4	k		s	20
w			110		
y	4	k	200		

IV. Modèle Relationnel

■ Troisième FN (3FN)

- Permet d'éliminer les redondances dues aux DF transitives.
- **Une relation est en 3FN si et seulement si :**
 - Elle est en 2FN
 - Tout attribut n'appartenant à aucune clé ne dépend pas d'un attribut non-clé. Il doit être directement dépendant (ou non transitivement dépendant) des clés.

- Pour savoir donc si un schéma est en 3FN, on doit :

- Chercher toutes ses clés minimales
- En déduire les attributs A qui n'appartiennent à aucune clé minimale
- Regarder toutes les DF $X \rightarrow A$ de F^+ , avec A non inclus dans X (cas trivial), et tester pour chacune d'elle si X est une clé.

IV. Modèle Relationnel

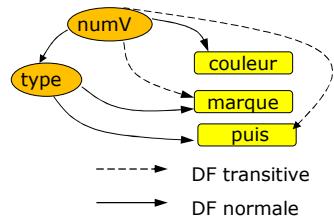
■ Remarques :

- Au lieu de considérer toutes les DF de F^+ , on se limiter à $\text{MIN}(F)$
- Pour montrer qu'un schéma R n'est pas en 3 FN, il suffit de donner une DF $X \rightarrow A$ de F^+ , avec X non-clé, A non inclus dans X et $\not\in$ à aucune clé.

■ Exemple :

Voiture(numV, marque, type, puis, couleur), avec
 $F=\{\text{numV} \rightarrow \text{type} ;$
 $\text{numV} \rightarrow \text{couleur} ;$
 $\text{numV} \rightarrow \text{marque} ;$
 $\text{numV} \rightarrow \text{puis} ;$
 $\text{type} \rightarrow \text{marque} ;$
 $\text{type} \rightarrow \text{puis}$
 $\}$

■ Graphe de DF



$\text{MIN}(F)=\{\text{numV} \rightarrow \text{type} ;$
 $\text{numV} \rightarrow \text{couleur} ;$
 $\text{type} \rightarrow \text{marque}, \text{puis}$
 $\}$

IV. Modèle Relationnel

■ La relation Voiture de clé **numV**,
n'est pas en 3FN car l'attribut
**non-clé type détermine les
attributs marque et puis.**

■ Du fait des DF transitives :

- Il a une redondance potentielle dans la relation Voiture avec des anomalies de stockage :
- Anomalie d'insertion, Anomalie de suppression, Anomalie de modification

Voiture

numV	couleur	type	marque	puis
6368CA07	rouge	R12TSE	Renault	6
6969CA07	bleue	R12TSE	Renault	6
4406BA04	verte	R25	Renault	11

IV. Modèle Relationnel

■ Les anomalies :

- **Anomalie d'insertion :**
On ne peut pas insérer une nouvelle occurrence du triplet (type, marque, puis) s'il n'existe pas une occurrence de numV constituant la clé primaire (contrainte d'intégrité d'entité).
- **Anomalie de suppression :**
Si l'on supprime une valeur de la clé primaire numV, on peut perdre une valeur unique du triplet (type, marque, puis) (ex. 4406BA04).

▪ **Anomalie de modification :**

Si on veut modifier du triplet (type, marque, puis) par exemple (R12TSE, Renault, 6), on aura soit un coût élevé de mise à jour (résultant de la redondance logique) soit une incohérence potentielle (si on effectue la mise à jour sur une seule occurrence).

IV. Modèle Relationnel

■ Décomposition de Voiture en 2 relations en 3FN :

- Le théorème de décomposition permet d'isoler les DF transitives et d'obtenir des relations en 3FN ne présentant pas les anomalies de stockage.
- $\text{MIN}(F) = \{\text{numV} \rightarrow \text{type}, \text{couleur} ; \text{type} \rightarrow \text{marque}, \text{puis}\}$

▪ Théorème de Heath :

$X = \text{type}, Y = \text{marque}, Z = \text{numV}, \text{couleur}$

Modele(type, marque, puis)
 $F = \{\text{type} \rightarrow \text{marque}, \text{puis}\}$

Voiture(numV, type, couleur)
 $F = \{\text{numV} \rightarrow \text{couleur}, \text{type}\}$

IV. Modèle Relationnel

■ Avantages de la 3FN

- Le fait de regrouper dans une relation, les attributs qui sont strictement dépendant de la clé permet de limiter les redondances.
- Considérons les deux schémas ci-dessous : On a 100 voitures et 30 modèles différents et chaque attribut est codé sur 1 octet.

Schémas	FN	Nombre d'octets
Voiture(<u>numV</u> , marque, type, puis, couleur)	2FN	100x5=500
Voiture(<u>numV</u> , type, couleur) Modèle(<u>type</u> , marque, puis)	3FN	100x3=300 30x3=90 300+90=390

IV. Modèle Relationnel

■ Exercice 1 :

- Soit le schéma $R(S, R, A, C)$ avec $F = \{S, A \rightarrow R ; S, R \rightarrow C\}$, avec **S = Succursale de magasin, R = un Rayon, A = un Article et C = Chef de rayon.**
- Montrer que **R n'est pas en 3FN et la décomposer.**

■ Exercice 2 :

- Montrer que la relation Etudiants ci-dessous n'est pas en 3FN et la décomposer.

Etudiants(numEt, nom, codeCl, libellé), avec
 $F = \{\text{numEt} \rightarrow \text{nom}, \text{codeCl}, \text{libellé} ;$
 $\text{codeCl} \rightarrow \text{libellé}\}$

IV. Modèle Relationnel

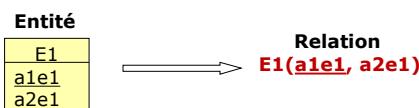
□ 4.9. Modèle Logique de Données Relationnelle

- MLDR
- Règles de passage
- Tableau de conversion

Entité	Relation
Propriété de l'entité	attribut de la relation
Identifiant de l'entité	Clé primaire de la relation

■ Règle1 : chaque entité devient une relation.

- L'identifiant de l'entité est la clé de la relation.



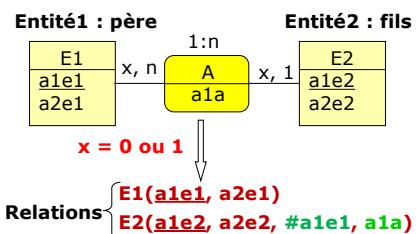
INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

89

■ Règle 2 : Association de type 1:n (relation de type père-fils)

- Le père est du côté de la cardinalité max n et le fils du côté de la cardinalité max 1.
- L'identifiant du père et les propriétés de l'association migrent chez le fils.

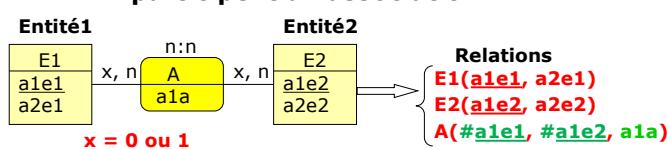


89

IV. Modèle Relationnel

■ Règle 3 : Association de type n:n

- L'association devient une relation.
- Ses propriétés sont les propriétés de l'association et les identifiants des entités qui participent à l'association.
- Sa clé est la concaténation des identifiants des entités qui participent à l'association.

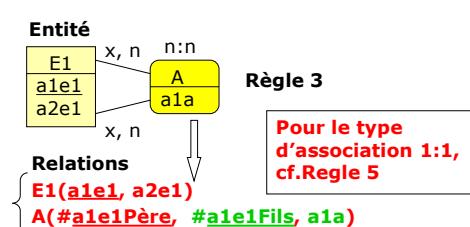
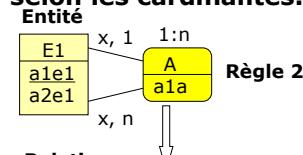


INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

■ Règle 4 : association réflexive

- On applique les règles 2 et 3 selon les cardinalités.



90

IV. Modèle Relationnel

■ Règle 5 : Association de type

1:1

Entité1
E1
a1e1
a2e1

Entité1
E1
a1e1
a2e1

Entité1
E1
a1e1
a2e1

A
a1a

A
a1a

A
a1a

Entité2
E2
a1e2
a2e2

Entité2
E2
a1e2
a2e2

Entité2
E2
a1e2
a2e2

1, 1
0, 1
0, 1

1, 1
0, 1
0, 1

1, 1
0, 1
0, 1

Relations

E1(a1e1, a2e1, #a1e2, a1a)
E2(a1e2, a2e2)
ou
E1(a1e1, a2e1)
E2(a1e2, a2e2, #a1e1, a1a)
ou
E1(a1e1, a2e1, #a1e2, a2e2, a1a)
ou
E2(a1e2, a2e2, #a1e1, a2e1, a1a)

Relations

E1(a1e1, a2e1, #a1e2, a1a)
E2(a1e2, a2e2)

Relations

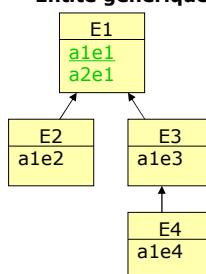
E1(a1e1, a2e1)
E2(a1e2, a2e2)
A(#a1e1, #a1e2, a1a)

IV. Modèle Relationnel

■ Règle 6 : héritage d'entité

- L'identifiant de l'entité générique devient la clé des entités spécialisées

Entité générique



Relations

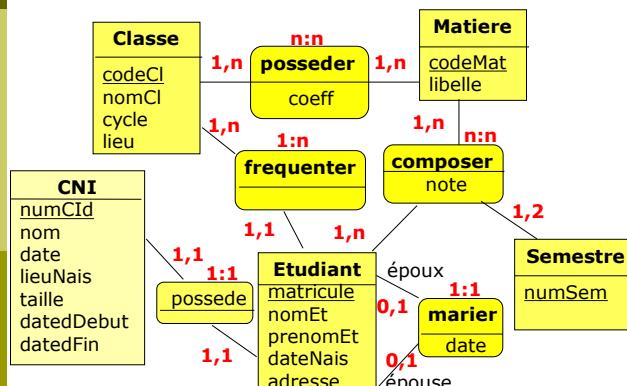
E1(a1e1, a2e1)
E2(a1e1, a1e2)
E3(a1e1, a1e3)
E4(a1e1, a1e4)

Entités spécialisées

IV. Modèle Relationnel

■ Exemple :

□ Schéma relationnel



□ Application des règles 1, 2, 3 et 5

```

Classe(codeCl, nomCl, cycle, lieu)
Etudiant(matricule, nomEt, prenomEt, dateNais,
adresse, #codeCl, #numCID)
Matiere(codeMat, libelle)
Semestre(numSem)

CNI(numCId, dateNais, lieuNais, taille, dateCIDebut,
dateCIFin)

Posseder(#codeCl, #codeMat, coeff)

Composer(#matricule, #codeMat, #numSem, note)

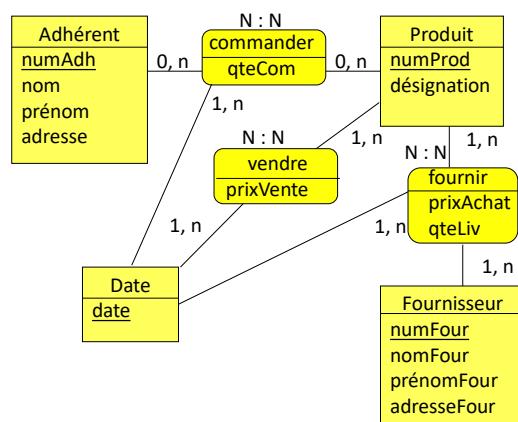
Marier(#matriculeEpoux, #matriculeEpouse, date)

```

IV. Modèle Relationnel

■ Exercice 1 :

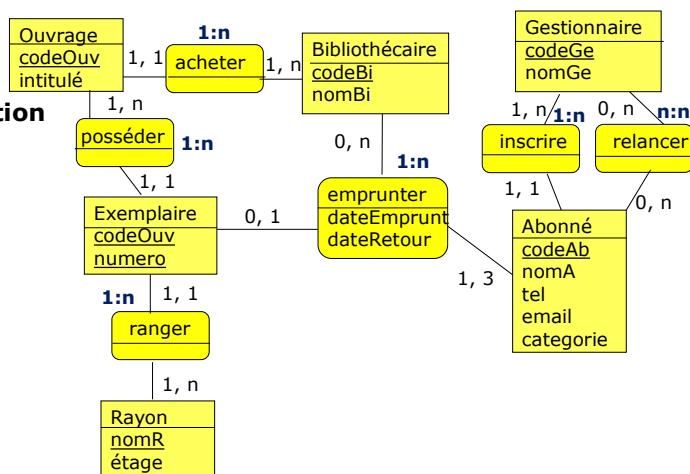
□ Trouver le schéma relationnel correspondant à ce schéma Entité/Association



IV. Modèle Relationnel

■ Exercice 2 :

- Trouver le schéma relationnel correspondant à ce schéma Entité/Association



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

95

95

V. SGBD Access

□ 5.1. Démarrage d'Access

- L'environnement de développement d'Access est celui de Windows.
 - Manipulation des objets liés à cet environnement graphique : souris, fenêtres, menus déroulant....
- Cliquer sur le bouton Démarrer
- Sélectionner Toutes les applications/Microsoft Access



- Cliquer sur l'icône "Base de données vide"

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

96

96

V. SGBD Access

The screenshot shows the Microsoft Access 2016 interface. A red arrow labeled '1' points to the 'Nom de fichier' field where 'BDGSF2016.accdb' is typed. Another red arrow labeled '2' points to the 'Créer' button at the bottom of the dialog box.

Donner un nom à la nouvelle) BD (ex BDGSF2016.accdb et choisir son répertoire de destination.

Cliquer sur le bouton Créer.

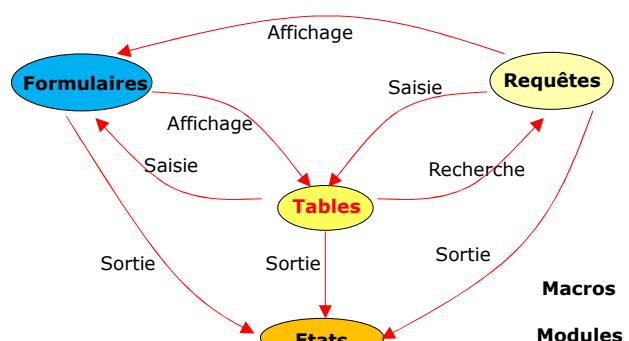
The ribbon is visible with the 'Créer' tab highlighted in blue. The window title is 'BDGSF2016 : Base de données - D:\Cours\Cours2014\BD\Access\Source\'. The 'Tables' section shows 'Table1'.

97

V. SGBD Access

5.2. Les éléments de base

- Une BD Access 2007 et + a l'extension .accdb
 - **Access DataBase**
 - **Elle regroupe dans un seul fichier les objets suivants :**



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

98

98

V. SGBD Access

■ Table :

- Les données d'une BD structurées en enregistrements sont réparties entre plusieurs contenants appelés tables ;
- chaque relation est représentée par une table (extension d'une relation).

■ Formulaire :

- Interface de visualisation ou de saisie de données dans les tables.

■ Requête :

- Elles permettent de rechercher et de récupérer les données stockées sans les tables répondant à certains critères.

■ Etats :

- Ils servent à analyser et à imprimer les données stockées dans les tables ou les requêtes.

■ Macros :

- Commandes permettant d'automatiser certaines tâches (ex. création et impression de bons de livraison et de factures).

■ Modules :

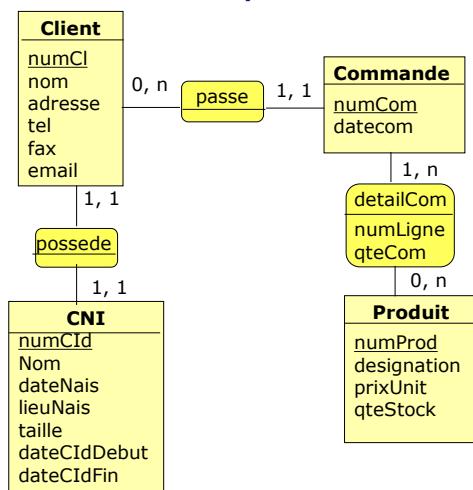
- Procédures permettant d'automatiser certaines tâches plus complexes non réalisables avec les macros.

VI. Les tables

□ 6.1. Schéma relationnel

- BD permettant de gérer les commandes.
- Règle de gestion
 - Un client peut passer ou non au moins une commande (0, n) ;
 - Un client possède une seule CNI (1, 1) ;
 - Une CNI appartient à un seul client (1, 1) ;
 - Une commande est passée par un et un seul client (1, 1).
 - Une commande contient plusieurs lignes de produits (1, n),
 - Un produit peut figurer ou non dans plusieurs commandes (0, n).

■ Schéma entité/association



VI. Les tables

■ Schéma relationnel

```

Client(numCl, Nom, adresse, tel, fax, email, #numCId)
Commande(numCom, datecom, #numCl)
Produit(numProd, designation, prixUnit, qteStock)
CNI(numCId, Nom, dateNais, lieuNais, taille, dateCIdDebut,
dateCIdFin)
DetailCom(#numCom, #numProd, numLigne, qteCom)
  
```

■ Structure des tables

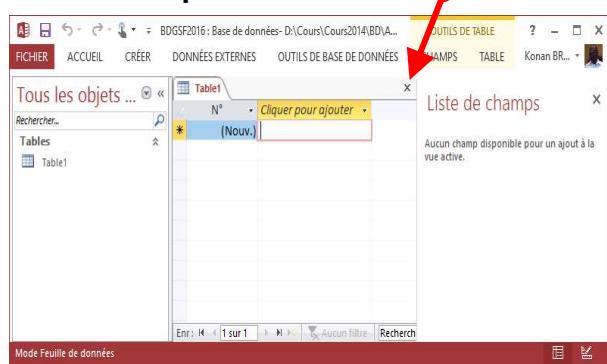
Tables	Champ	Type	Taille
CNI	numCId	Texte cours	15 caractères
	nom	Texte court	20 caractères
	dateNais	Date/Heure	Date abrégée
	lieuNais	Texte court	15 caractères
	taille	Numérique	Octet
	dateCIdDebut	Date/Heure	Date abrégée
	dateCIdFin	Date/Heure	Date abrégée

Tables	Champ	Type	Taille
Client	numCl	Numérique	Entier long
	nom	Texte court	20 caractères
	adresse	Texte court	30 caractères
	tel	Numérique	Entier long
	fax	Numérique	Entier long
	email	Texte court	50 caractères
Produit	numCId	Texte court	15 caractères
	numProd	Numérique	Entier long
	designation	Texte court	15 caractères
	prixUnit	Numérique	Réel simple
	qteStock	Numérique	Entier
	numCom	Numérique	Entier long
Commande	dateCom	Date/Heure	Date abrégée
	numCl	Numérique	Entier long
	numCom	Numérique	Entier long
DetailCom	numProd	Numérique	Entier long
	numLigne	NuméroAuto	Entier long
	qteCom	Numérique	Entier

VI. Les tables

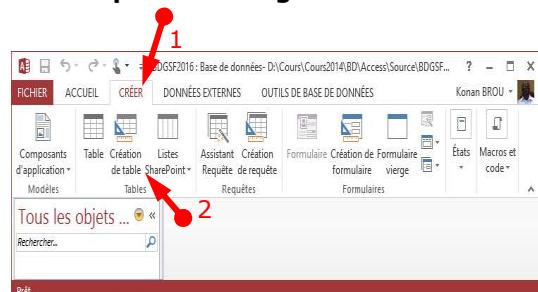
□ 6.2. Crédation de la table client

- Fermer la fenêtre affichée
- Cliquer sur l'icône Fermeture



■ Onglet CREER

- Renferme toutes les commandes de création des objets d'Access.
- Cliquer sur l'onglet CREER



- Cliquer sur l'icône "Création de table"

VI. Les tables

Ci-dessus l'interface de création des tables

Cliquer dans Nom du champ et saisir numCl

Cliquer dans "Type de données" et sélectionner "Numérique"

Cliquer dans "Taille du champ" et sélectionner "Entier long"

Cliquer sur l'icône Clé primaire

103

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

103

VI. Les tables

- Chaque champ est spécifié sur une ligne du tableau.
- Pour chaque champ, répéter les opérations suivantes :
 1. Taper le nom du champ dans la première colonne (Nom du champ)
 2. Cliquer dans la zone "Type de données" et choisir un des types proposés dans la liste déroulante
 - Texte, Mémo, Numérique, Date/Heure, Monétaire, NuméroAuto, Oui/Non, Objet OLE, Lien hypertexte, Assistant liste de choix.
- 3. Taper dans la colonne "Description" un commentaire explicatif sur le champ (facultatif).
- 4. Préciser les différentes propriétés du champ (partie inférieure de la fenêtre).
 1. Les propriétés sont :

104

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

104

VI. Les tables

Propriétés	Explication
Taille du champ	capacité de contenance du champ (longueur du texte...).
Format	format des valeurs du champ (ex. pourcentage, monétaire).
Décimale	nombre de chiffres décimaux dans.
Masque de saisie	forme sous laquelle la valeur du champ doit être saisie.
Légende	libellé du champ qui doit apparaître dans un formulaire.
Valeur par défaut	valeur automatiquement insérée dans le champ pour les nouveaux enregistrements.
Valide si	condition qui limite les valeurs prises par le champ
Message si erreur	message à afficher si on entre une valeur non permise.
Nul interdit	si oui alors le champ doit obligatoirement contenir une valeur pour chaque enregistrement (ex. valeur d'une clé).
Chaîne vide autorisée	si oui, la chaîne vide est autorisée pour les champs de type Texte
Indexé	permet d'accélérer les recherches et les tris, mais ralentit les mises à jour.

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

105

5. Indiquer la ou les clé(s) de la table

- sélectionner la ou les ligne(s) et cliquer sur l'icône clé primaire

6. Enregistrer la table

7. Fermer la table

105

VI. Les tables

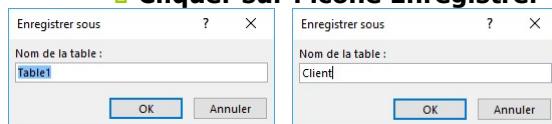


■ NB :

- Eviter les espaces et lettres accentuées dans le nom des tables et des attributs.

■ Sauvegarder la table

- Cliquer sur l'icône Enregistrer



- Donner le nom de la table (Client) et cliquer sur le bouton OK

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

106

106

VI. Les tables

□ Suivre ces différentes étapes pour créer les autres tables

Tables	Champ	Type	Taille
Produit	numProd	Numérique	Entier long
	designation	Texte court	15 caractères
	prixUnit	Numérique	Réel simple
	qteStock	Numérique	Entier
	numCom	Numérique	Entier long
	dateCom	Date/Heure	Date abrégée
Commande	numCl	Numérique	Entier long

Tables	Champ	Type	Taille
CNI	numCId	Numérique	Entier long
	nom	Texte court	20 caractères
	dateNais	Date/Heure	Date abrégée
	lieuNais	Texte court	15 caractères
	taille	Numérique	Octet
	dateCIdDebut	Date/Heure	Date abrégée
	dateCIdFin	Date/Heure	Date abrégée

INP-HBI/K. M. BROU Chapitre 1 : Access niveau 1 107

107

VI. Les tables

■ Remarque

- Pour la table **DetailCom**, sélectionner les deux lignes **numCom** et **numProd** avant de cliquer sur l'icône "Clé primaire"

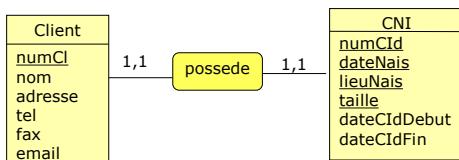
Tables	Champ	Type	Taille
DetailCom	numCom	Numérique	Entier long
	numProd	Numérique	Entier long
	numLigne	NuméroAuto	Entier long
	qteCom	Numérique	Entier

108

VI. Les tables

□ 6.3. Relations entre les tables

- Permet de vérifier les contraintes d'intégrités référentielles.
- Relation de type 1 : 1
 - A un enregistrement de la table source, ne correspond qu'un seul enregistrement de la table destination et vice versa.



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

109

CarteIdentite	
Nom du champ	Type de données
numCId	Texte court
nom	Texte court
dateNais	Date/Heure
lieuNais	Texte court
dateDebut	Date/Heure
dateFin	Date/Heure

CNI(**numCId**, nom, dateNais, lieuNais, dateDebut, dateFin)
Client(**numCl**, nom, adresse, tel, fax, email, **#numCId**)

109

VI. Les tables

The screenshot shows the Microsoft Access object browser with "Tables" selected. The "Client" table is highlighted. In the main pane, the "Relations" tab is active, showing a relationship named "Client" connecting the "Client" and "CNI" tables. The "Properties" tab is open for the "numCl" field, which is defined as a "Numérique" type with a size of 15. The "General" tab displays various properties like "Format" and "Masque de saisie". The "Liste de choix" tab shows that the "Indexé" (Indexed) option is set to "Oui-Sans doublon" (Yes-Without duplicates). The status bar at the bottom indicates "Mode Création. F6 = Autre volet. F1 = Aide."

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

110

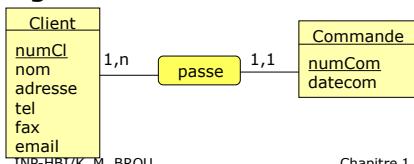
110

- Ouvrir la table Client en mode création
- Cliquer sur le champ numCId
- Dérouler la liste "Indexé" et sélectionner "Oui-Sans doublon"
- Enregistrer et fermer les tables ouvertes

VI. Les tables

■ Relation de type 1 : N

- Deux tables sont reliées par l'intermédiaire d'une clé primaire (table source) et d'une clé externe (table destination) ;
- à un enregistrement de la table source, peuvent correspondre plusieurs enregistrements ;
- à l'inverse, à un enregistrement de la table destination, ne peut correspondre qu'un seul enregistrement de table source.



Chapitre 1 : Access niveau 1

111

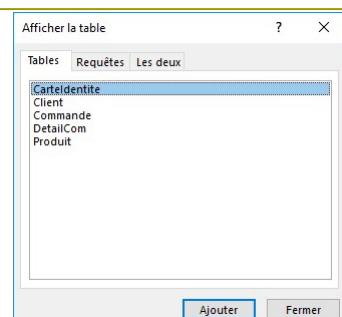
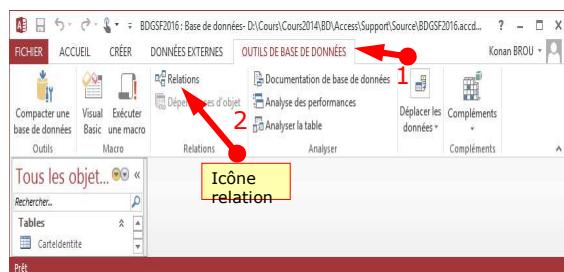
- Client(numCl, nom, adresse tel, fax, email)
- Commande(numCom, dateCom, #numCl)
- Relation de type 1 : N entre Client et Commande
- Relation de type 1 : N entre Commande et DetailCom
- Relation de type 1 : N entre Produit et DetailCom

111

VI. Les tables

■ Création des relations

- Cliquer sur l'onglet "OUTIL DE BASE DE DONNEES"



- Ajouter successivement les 5 tables et fermer la fenêtre Ajout de tables

- Cliquer sur l'icône Relations

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

112

112

VI. Les tables



- Glisser le champ numCId de la table CarteIdentite sur numCId de la table Client ou vice versa.

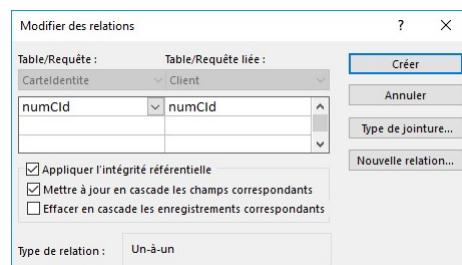


INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

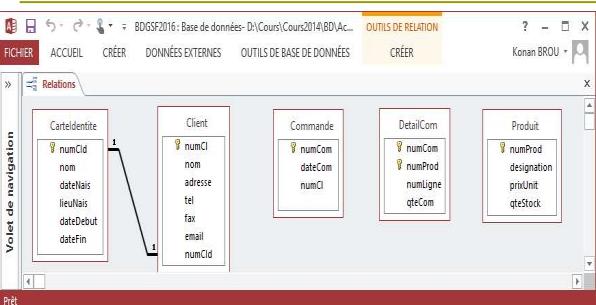
113

- Cocher la case "Appliquer l'intégrité référentielle"
- Cocher la case "Mettre à jour en cascade les champs correspondants" et cliquer sur le bouton Créer.

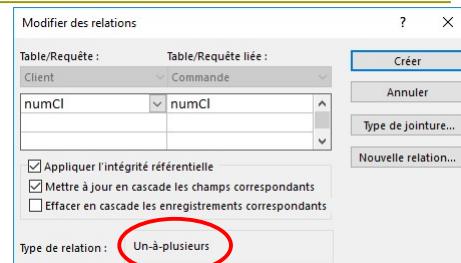


113

VI. Les tables



- Glisser le champ numCId de la table Client sur numCId de la table Commande.



- Cocher la case "Appliquer l'intégrité référentielle" ainsi que la case "Mettre à jour en cascade les champs correspondants" et cliquer sur le bouton Créer.

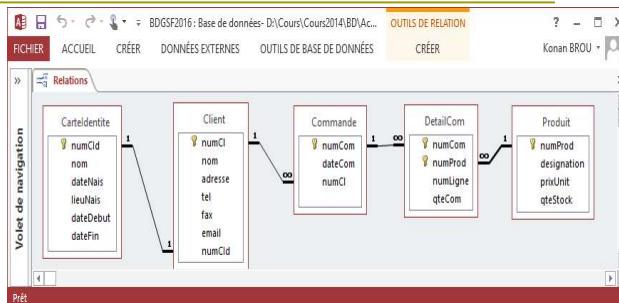
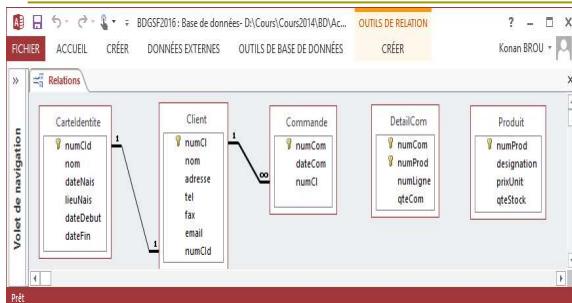
INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

114

114

VI. Les tables



■ De même

- Glisser le champ numCom de la table Commande sur numCom de la table DetailCom.
 - Glisser le champ numProd de la table Produit sur numProd de la table DetailCom.

□ Enregistrer

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

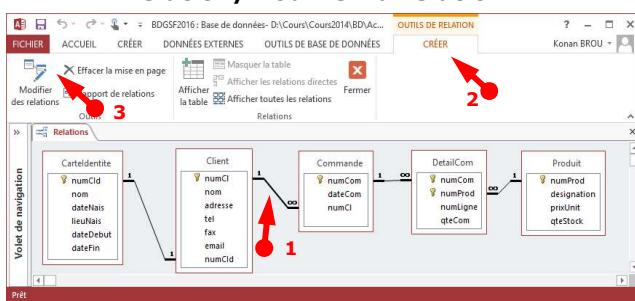
115

115

VI. Les tables

■ Modification d'une relation

- Cliquer sur la relation
 - Cliquer sur l'onglet "CRÉER" , puis sur l'icône "Modifier des relations"
 - Ou clic droit sur la relation /Modifier la relation



Modifier des relations

Table/Requête :	Table/Requête liée :	OK
Client	Commande	Annuler
numCl	numCl	Type de jointure...
<input checked="" type="checkbox"/> Appliquer l'intégrité référentielle <input checked="" type="checkbox"/> Mettre à jour en cascade les champs correspondants <input type="checkbox"/> Effacer en cascade les enregistrements correspondants		Nouvelle relation...
Type de relation :		Un-à-plusieurs

□ Cocher les options désirées

INP-HBI/K. M. BROU

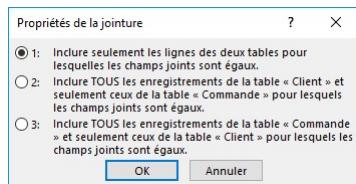
Chapitre 1 : Access niveau 1

116

116

VI. Les tables

- Cliquer sur le bouton Type jointure pour modifier les types de jointure



- Sélectionner une option est valider
- Enregistrer les relations et fermer la fenêtre "Relations"

VI. Les tables

□ 6.4. Saisie des données

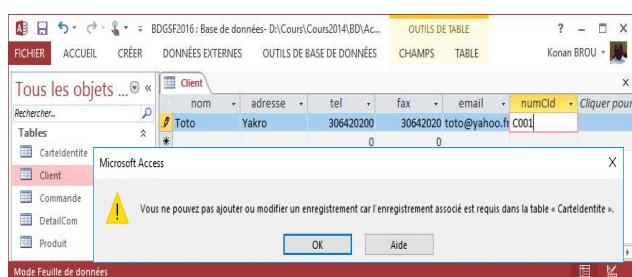
- Double cliquer sur la table Client ou faire un clic droit sur la table Client et sélectionner Ouvrir
- Saisir les enregistrements suivants :

Client						
numCl	nom	adresse	tel	fax	email	numCId
1	Toto	Yakro	30642020	30642020	toto@yahoo.fr	C001
2	Frotomougou	Abidjan	22343434	22343434		C002
3	Koffi Adjoua	Bouaké	30636565	30636565		C003
4	Seri	Daloa	34343434	34343434	seri@gmail.com	C004
5	Koffi	Yakro	0	0		C005

VI. Les tables

■ Access génère l'erreur suivante :

- "Vous ne pouvez ajouter ou modifier un enregistrement car l'enregistrement associé est requis dans la table CarteIdentite"



□ Cette erreur est due à la contrainte d'intégrité référentielle placée sur le champs numCId.

□ Annuler l'enregistrement et fermer la table Client (Appuyer sur la touche echap).

119

VI. Les tables

- Double cliquer sur la table CarteIdentie
- Saisir les enregistrements suivants :

CarteIdentite						
numCId	nom	dateNaiss	lieuNaiss	Taille	dateDebut	dateFin
C001	Toto	21/04/1962	Paris	188	30/04/2016	29/04/2026
C002	Frotomougou	25/04/1962	Yakro	159	30/04/2016	29/04/2026
C003	Koffi Adjoua	31/01/1965	Yakro	178	03/07/2014	02/07/2024
C004	Seri	14/06/1998	Tipatipa	120	30/04/2016	29/04/2026
C005	Koffi	22/05/1996	Man	165	10/03/2015	09/03/2025

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

120

120

VI. Les tables

- Fermer la table CarteIdentite
- Saisir à nouveau les enregistrements de la table Client

■ De même, saisir dans la table Produit les enregistrements suivants :

Produit			
numProd	designation	prixUnit	qteStock
1	Savon	300	15
2	Eau Awa	400	29
3	Sardine	450	10
4	Assiette	245	27
5	Yaourt	250	50

VI. Les tables

- De même, saisir dans les tables Commande et DetailCom les enregistrements suivants :

Commande		
numCom	dateCom	numCI
1	13/11/2014	1
2	12/05/2014	1
3	13/11/2014	1
4	15/05/2015	2
5	18/05/2015	2
6	18/11/2015	3
7	05/04/2015	3
8	06/05/2016	3
9	06/05/2016	4
10	12/05/2016	4

DetailCom		
numCom	numProd	qteCom
1	1	10
1	2	4
2	2	3
3	1	2
3	2	6
3	3	3
4	2	7
5	2	3
5	3	6
6	2	1
7	3	5
8	2	8
9	1	3
10	2	4

VII. Les requêtes

□ 7.1. Présentation

- Une requête permet de rechercher des enregistrements répondant à certains critères et d'en établir la liste.
- Une requête peut contenir une ou plusieurs tables.
 - Toute modification dans une de ces tables est automatiquement répercutée sur la requête.
 - C'est la raison pour laquelle les requêtes sont appelées feuille de réponse dynamique (Dynaset).

■ Deux façons de créer les requêtes :

- mode graphique : Access fournit une interface de type QBE (Query By Example) permettant de créer des requêtes de façon visuelle ;
- mode SQL : saisie directe des commandes SQL.

VII. Les requêtes

■ Interface de requêtes

- Cliquer sur l'onglet Créer



VII. Les requêtes

■ Langage SQL : Structured Query Language

- Langage ensembliste non procédurale pour la manipulation d'une BD relationnelle.
- Basé sur l'algèbre relationnelle.
- Inventé par IBM en 1982.
- Il a fait l'objet de plusieurs normes ANSI/ISO depuis 1987.
 - La norme la plus répandue est SQL2 définie en 1992.

□ Intègres plusieurs niveaux de langage :

- Langage de Définition des Données (LDD) : **ALTER, CREATE, DROP**
- Langage d'Interrogation de Données (LID) : **SELECT**
- Langage de Modification de Données (LMD) : **UPDATE, INSERT, DELETE**
- Langage de Contrôle des Données et des utilisateurs (LCD) : **GRANT, REVOKE**

VII. Les requêtes

■ Langage d'Interrogation de Données (LID)

- Consiste à rechercher des données contenues dans des tables.
- Syntaxe :

```
SELECT [ALL|DISTINCT] attribut(s)
FROM nomTable(s)
[WHERE condition]
[GROUP BY attributs]
[HAVING condition]
[ORDER BY attribut(s) [ASC|DESC]] ;
```

VII. Les requêtes

□ 7.2. Requêtes sélection

■ Consultation simple

▢ Syntaxe :

```
SELECT ListeDeSélection  
FROM ListeDeTable;
```

▢ ListeDeSélection : permet de spécifier les colonnes à visualiser.

- Les noms de colonnes seront séparées par une virgule.

▢ ListeDeTable : permet de spécifier le nom de la table qui contient les données à sélectionner.

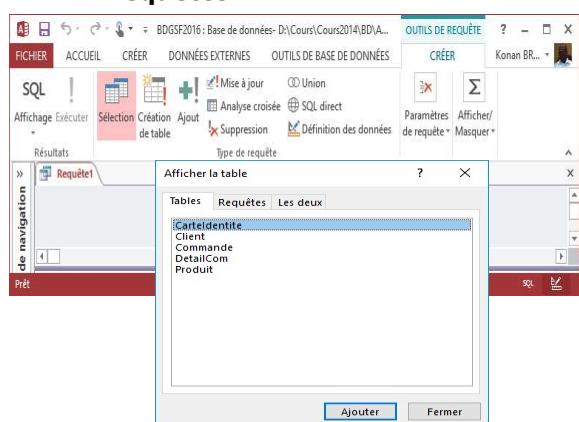
▢ Exemple 1 : Extension d'une table (toutes les colonnes)

▪ Requête 1 : Liste de tous les clients

```
SELECT *  
FROM Client;
```

VII. Les requêtes

▢ Cliquer sur l'icône "Création de requêtes"



▢ Cliquer la table Client et cliquer sur le bouton Ajouter

▢ Cliquer su le bouton Fermer

VII. Les requêtes

The left screenshot shows the 'Requête1' query design view. A red callout points to the 'Client' table in the 'Champ:' dropdown, with the text 'Représente tous les champs' (Represents all fields) overlaid. The right screenshot shows the same view with a red circle around the 'Exécuter' (Run) button in the toolbar.

Double cliquer sur le champ *

Exécuter la requête en cliquant sur l'icône Exécuter !

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

129

129

VII. Les requêtes

The screenshot shows the 'Requête1' query results view. A red callout points to the 'Enregistrer' (Save) button in the toolbar. To the right, two 'Enregistrer sous' (Save As) dialog boxes are shown, one for 'Nom de la requête:' (Query name) and another for 'Nom de la requête:' (Query name) with the value 'RListeClient1' entered.

On a toutes les données de la table Client

Enregistrer la requête sous le nom : RListeClient1

- Cliquer l'icône Enregistrer et saisir RListeClient

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

130

130

VII. Les requêtes

Icône Affichage

Visualiser le code SQL générée de RListeClient1

- Dérouler la liste "Affichage"

Un objet Requête a été ajouter

Sélectionner "SQL Mode SQL"

INP-HBI/K. M. BROU Chapitre 1 : Access niveau 1 131

131

VII. Les requêtes

Remarque : une requête peut être :

- Renommer
- Modifier
- Réexécuter
- Supprimer

Fermer la requête RListeClient

INP-HBI/K. M. BROU Chapitre 1 : Access niveau 1 132

132

VII. Les requêtes

Exemple 2 : limiter le nombre de colonnes

- Dans la pratique, il est rare que l'on souhaite avoir toutes les colonnes d'une table, on se limite le plus souvent à des colonnes.

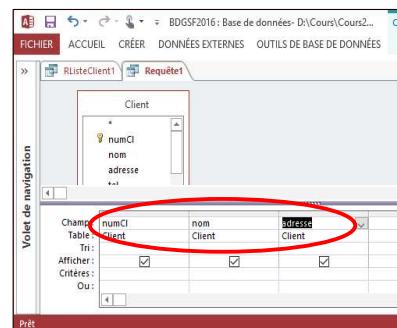
Syntaxe

```
SELECT attribut1, attribut2, ...  
FROM nomTable;
```

Requête 2 : numéro, nom et adresse des clients.

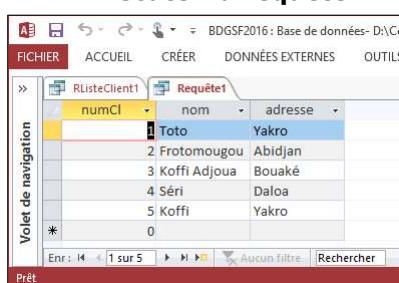
```
SELECT numCl, nom, adresse  
FROM Client;
```

- Créer une nouvelle requête basée sur la table Client
- Double cliquer sur les champs numCl, nom et adresse



VII. Les requêtes

Exécuter la requête



Enregistrer la requête sous le nom RListeClient2

```
SELECT Client.numCl, Client.nom, Client.adresse  
FROM Client;
```

Fermer la requête

VII. Les requêtes

Exemple 3 : Projection

- Suppression de certaines colonnes avec élimination des enregistrements en double.
- On utilise la clause **DISTINCT**
- Syntaxe

```
SELECT DISTINCT attribut1, attribut2, ...  
FROM nomTable;
```

Requête 3 : Liste de toutes les villes

- Sélection avec des doublons

```
SELECT adresse  
FROM Client;
```

- Sélection sans doublons

```
SELECT DISTINCT adresse  
FROM Client;
```

- Créer une nouvelle requête basée sur la table Client
- Double cliquer sur les champs adresse
- Exécuter la requête (1)
- Passer en mode SQL et ajouter DISTINCT après le SELECT (2)
- Exécuter la requête (3)

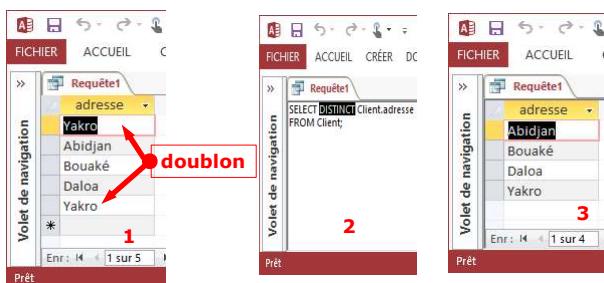
INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

135

135

VII. Les requêtes



- Enregistrer la requête sous le nom RListeVille
- Fermer la requête RListeVille

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

136

136

VII. Les requêtes

■ Consultation avec restriction

- En général, les requêtes ne portent pas sur la totalité d'une table, seules quelques lignes sont sélectionnées selon une certaine condition (restriction).
- Cette condition s'exprime dans la clause WHERE.
- Syntaxe

SELECT col ₁ , col ₂ , col _n FROM table WHERE conditions;	Syntaxe de la condition attribut opérateur valeur
--	---

□ Opérateurs de comparaison

Opérateur	Signification
=	égalité
<>	différence
<, <=, >, >=	infériorité et supériorité
IS NULL	teste si le contenu d'une cellule est nulle
IN(liste_de_valeur)	égalité à un élément d'une liste
BETWEEN x AND y	comparaison par rapport à un intervalle de valeurs
EXIST	existence d'une ligne de résultats
NOT, OR, AND :	opérateurs logiques
LIKE chaîne générique	égalité à une sous-chaîne dans la valeur d'un champ

VII. Les requêtes

□ Remarque

- On peut avoir également la négation de ces opérateurs :
- IS NOT NULL, NOT IN, NOT BETWEEN, NOT LIKE

□ Condition

```
exp opérateurRelationnel exp
exp [NOT] BETWEEN exp AND exp
exp [NOT] IN (listeDeValeurs)
colonne [NOT] LIKE "chaîne"
colonne IS [NOT] NULL
```

□ exp désigne une expression qui peut être :

- un nom de colonne
- une constante numérique ou caractère
- une pseudo colonne
- une valeur nulle
- ou une combinaison de ces éléments par des opérateurs logiques.

VII. Les requêtes

Exemple 1 : Restriction simple

Requête 4 : Liste des clients de Yakro.

- Tous les clients ont été sélectionnés dans la requête 2.
- Ici, on souhaiterait avoir uniquement les clients de Yakro.

```
SELECT numCl, nom, adresse
FROM Client
WHERE adresse = 'Yakro';
```

- Créer une nouvelle requête basée sur la table Client
- Ajouter les champs : numCl, nom et adresse

Champ : numCl nom adresse
Table : Client Client Client
Tri :
Afficher :
Critères :
Où : Yakro

- Saisir Yakro à l'intersection de la ligne Critères et la colonne adresse
- Exécuter la requête

139

Chapitre 1 : Access niveau 1

139

VII. Les requêtes

numCl	nom	adresse
Koffi	Yakro	Yakro
1 Toto	Yakro	Yakro
*		

Enregistrer la requête sous le nom RListeClientYakro

Requête 5 : Liste des produits dont la quantité en stock est inférieure 15.

```
SELECT numProd, designation, qteStock
FROM Produit
WHERE qteStock <= 15;
```

- Créer une nouvelle requête basée sur la table Produit
- Ajouter les champs : numProd, designation et qteStock

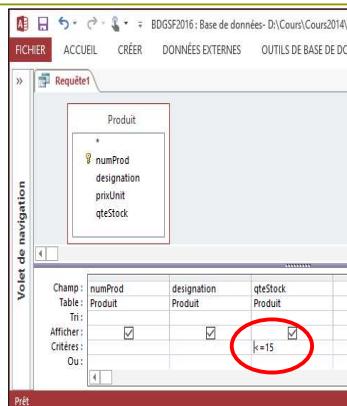
140

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

140

VII. Les requêtes



- Saisir ≤ 15 à l'intersection de la ligne Critères et la colonne qteStock
- Exécuter la requête

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

numProd	designation	qteStock
1 Savon		15
3 Sadine		10
*		0

- Enregistrer la requête sous le nom RListeProduits
- Visualiser le code SQL généré

```
SELECT Produit.numProd, Produit.designation,  
Produit.qteStock  
FROM Produit  
WHERE (((Produit.qteStock)<=15));
```

- Fermer la requête

141

141

VII. Les requêtes

■ Exercices :

- Ecrire la requête SQL pour avoir le Numéro, nom et tel du client Toto.
- Ecrire la requête SQL pour avoir le numéros et le nom des client de Yakro ou d'Abidjan.
- Ecrire la requête SQL pour avoir la liste des commandes du client numéro 2.
- Ecrire la requête SQL pour avoir le numéro et le nom des produits dont le prix est à inférieur à 200.

- Ecrire la requête SQL pour avoir le numéro et le nom des produits dont le prix est compris entre 200 et 300 (2 solutions).

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

142

142

VII. Les requêtes

Exemple 2 : Requête paramétrée

- La valeur de la condition est une variable
- **Requête 4 : Liste des clients d'adresse x**
- Utilisation d'une variable comme adresse du client
- Créer une nouvelle requête basée sur la table Client
- Ajouter les champs numCl, nom et adresse
- Taper dans la zone Critère du champ adresse
 - **= [Entrez l'adresse du client]**

Champ : numCl
Table : Client
Tri :
Afficher :
Critères :
Où : [=Entrez l'adresse du client]

- Exécuter la requête
- Saisir l'adresse désirée et cliquer



143

143

VII. Les requêtes

- Enregistrer la requête sous le nom RListeClient4
- Visualiser le code SQL généré

```
SELECT Client.numCl, Client.nom, Client.adresse  
FROM Client  
WHERE (((Client.adresse)=[Entrer l'adresse du client]));
```

- Fermer la requête

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

144

144

VII. Les requêtes

■ Le prédictat LIKE

- Permet de réaliser une comparaison entre la valeur de la colonne et celle d'une chaîne en utilisant des caractères génériques de substitution.
- Syntaxe
 - colonne [NOT] LIKE "chaine" [ESCAPE 'c']

□ ESCAPE :

- Permet d'identifier un caractère comme une séquence escape.
- Ceci est surtout utilisé pour permettre l'utilisation des caractères génériques comme des caractères littéraux.
- Si la séquence d'escape est définie par le caractère "\ alors "\ signifie que le souligné est une donnée et non pas un caractère générique.

□ NB : Dans Access version française LIKE = Comme

VII. Les requêtes

□ Les caractères génériques

- Utilisés dans des requêtes et des expressions pour inclure tous les enregistrements, dont l'intitulé commence par certains caractères ou correspond à une séquence particulière.

□ Remarque

- Les caractères génériques peuvent être considérés comme des caractères ordinaires dans une expression uniquement s'ils sont entourés de crochets.

Symbol	Utilisation	Exemple
*	un nombre quelconque de caractères et peut être placé en tout point de la chaîne de caractères.	Wh* trouve what, white et why, *at trouve cat, bat et what
?	un seul caractère.	B?ll trouve ball, bell et bill
#	un seul chiffre.	1#3 trouve 103, 113, 123
[]	un seul caractère parmi ceux indiqués entre crochets.	b[ae]ll trouve ball et bell mais pas bill
!	tout caractère ne figurant pas entre crochets.	b[!ae]ll trouve bill et bull mais pas bell ni ball
-	l'un des caractères de la plage indiquée.	b[a-c]d trouve bad, bbd et bcd

VII. Les requêtes

- Exemple 1 : Clients (numCl, nom et adresse) dont le nom se termine par "to".

```
SELECT numCl, nom, adresse  
FROM Client  
WHERE nom LIKE '*to';
```

Champ :	numCl	nom	adresse
Table :	Client	Client	Client
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :		Comme *to'	
Ou :			

- Exemple 2 : Clients (numCl, nom et adresse) dont le nom contient "to".

```
SELECT numCl, nom, adresse  
FROM Client  
WHERE nom LIKE '*to*';
```

- Exemple 3 : Clients (numCl, nom et adresse) dont le 2^e caractère du nom est r.

```
SELECT numCl, nom, adresse  
FROM Client  
WHERE nom LIKE '?r*';
```

- Exemple 4 : Clients (numCl, nom et adresse) dont le 3^e caractère du nom est "t".

```
SELECT numCl, nom, adresse  
FROM Client  
WHERE nom LIKE '??t*';
```

Champ :	numCl	nom	adresse
Table :	Client	Client	Client
Tri :			
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :		Comme ??	
Ou :			

VII. Les requêtes

7.3. Requêtes jointures

- Une jointure permet de récupérer dans une requête des données disséminées dans deux tables
 - C'est un lien entre deux tables disposant d'au moins une colonne commune (clé étrangère).
- L'opération de jointure consiste à créer une table temporaire composée de lignes satisfaisant la condition de jointure.
 - C'est en fait un produit cartésien avec une restriction.

VII. Les requêtes

■ Syntaxe :

```
SELECT listeDeSélection  
FROM listeDeTables  
WHERE condition;
```

- **listeDeSélection : composée des colonnes que l'on veut voir apparaître dans la jointure.**
 - Les noms des colonnes peuvent être précédées des noms des tables ou d'alias lorsqu'il y a ambiguïté.

- **listeDeTables : composée des noms de tables à partir desquelles les données sont extraites.**

- les noms de tables sont séparées par une virgule.

- **La condition de recherche peut être une liste de comparaisons simples conduisant à un produit cartésien ou à une liste de comparaison entre les colonnes de différentes tables produisant une jointure.**

VII. Les requêtes

■ Exemple 1 :

- **Requête 5 : Liste des clients (numCl, nom, adresse) avec leur commande (numCom, dateCom).**

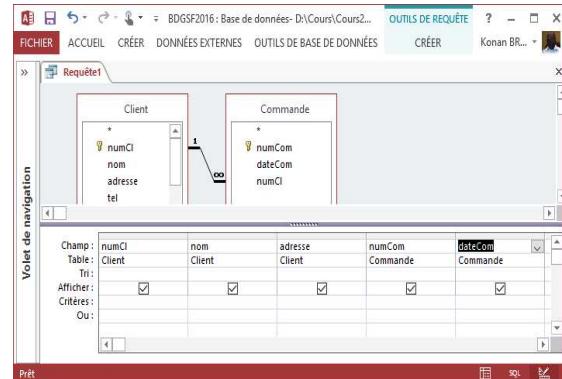
- Requête impliquant deux tables : Client et Commande

```
SELECT Cl.numCl, nom, adresse, numCom,  
dateCom  
FROM Client cl, Commande cd  
WHERE cl.numCl = cd.numCl;
```

Client		Commande		
numCl	nom	numCom	dateCom	numCl
1	Toto Ali	1	13/11/2003	1
2	Frotomogou	2	12/05/2004	1
3	Koffi Adjoua	3	13/02/2003	1
4	Seri	4	15/05/2004	2
5	Koffi	5	18/05/2004	2
6	Fatou	6	12/03/2004	3

VII. Les requêtes

- Créer une nouvelle requête
- Ajouter les tables Client et Commande
- Ajouter les champs : numCl, nom, adresse, numCom, dateCom



▪ La jointure est faite automatiquement grâce aux relations créées entre les tables.

□ Exécuter la requête

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

151

151

VII. Les requêtes

numCl	nom	adresse	numCom	dateCom
1 Toto	Yakro		1	13/11/2014
1 Toto	Yakro		2	12/05/2014
1 Toto	Yakro		3	13/11/2014
2 Frotomougou	Abidjan		4	15/05/2015
2 Frotomougou	Abidjan		5	18/05/2015
3 Koffi Adjoua	Bouaké		6	18/11/2015
3 Koffi Adjoua	Bouaké		7	05/04/2015
3 Koffi Adjoua	Bouaké		8	06/05/2016
4 Séri	Daloa		9	06/05/2016
4 Séri	Daloa		10	12/05/2016

```
SELECT Client.numCl, Client.nom, Client.adresse,
Commande.numCom, Commande.dateCom
FROM Client INNER JOIN Commande ON
Client.numCl = Commande.numCl;
```

□ Fermer la requête

- Enregistrer la requête sous le nom RCommandeClient1
- Visualiser le code SQL généré

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

152

152

VII. Les requêtes

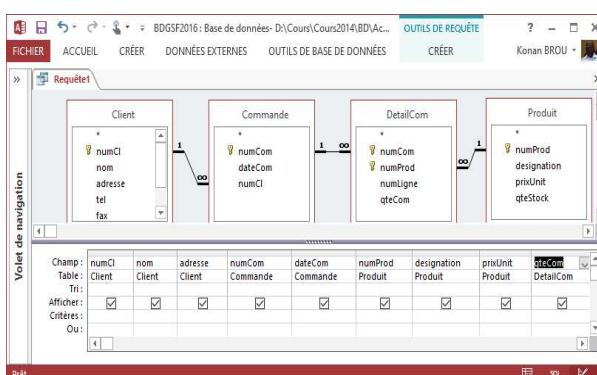
- **Requête 6 : Liste des clients (numCl, nom, adresse), numCom, dateCom, qteCom, désignation et prixUnit des produits qu'ils ont commandés.**

- Requête impliquant toutes les tables

```
SELECT cl.numCl, nom, cd.numCom, dateCom, pr.numProd,
       designation, prixUnit, qteCom
  FROM Client cl, Commande cd, DetailCom de, Produit pr
 WHERE cl.numCl = cd.numCl AND
       cd.numCom = de.numCom AND
       de.numProd = pr.numProd;
```

- Créer une nouvelle requête
- Ajouter les tables : Client, Commande, DetailCom et Produit
- Ajouter les champs : numCl, nom, adresse, numCom, dateCom, numProd, designation, prixUnit, qteCom

VII. Les requêtes



- **Exécuter la requête**

numCl	nom	adresse	numCom	dateCom	numProd	designation	prixUnit	qteCom
1 Toto	Yakro		1 13/11/2014		1 Savon		300	10
1 Toto	Yakro		2 12/05/2014		2 Eau Awa		400	4
1 Toto	Yakro		3 13/11/2014		1 Savon		300	2
1 Toto	Yakro		3 13/11/2014		2 Eau Awa		400	6
1 Toto	Yakro		3 13/11/2014		3 Sadine		450	3
2 Frotomougou	Abidjan		4 15/05/2015		2 Eau Awa		400	7
2 Frotomougou	Abidjan		5 18/05/2015		2 Eau Awa		400	3
2 Frotomougou	Abidjan		5 18/05/2015		3 Sadine		450	6
3 Koffi Adjoupa	Bouaké		6 18/11/2015		2 Eau Awa		400	1
3 Koffi Adjoupa	Bouaké		7 05/04/2015		3 Sadine		450	5
3 Koffi Adjoupa	Bouaké		8 06/05/2016		2 Eau Awa		400	8
4 Séri	Daloa		9 06/05/2016		1 Savon		300	3
4 Séri	Daloa		10 12/05/2016		2 Eau Awa		400	4

- **Enregistrer la requête sous le nom RCommandeClient2**

VII. Les requêtes

□ Visualiser le code SQL générée

```
SELECT Client.numCl, Client.nom, Client.adresse,
Commande.numCom, Commande.dateCom, Produit.numProd,
Produit.designation, Produit.prixUnit, DetailCom.qteCom
FROM Produit INNER JOIN ((Client
INNER JOIN Commande ON Client.numCl = Commande.numCl)
INNER JOIN DetailCom ON Commande.numCom =
DetailCom.numCom) ON Produit.numProd = DetailCom.numProd;
```

□ Fermer la requête

VII. Les requêtes

□ 7.4. Tri des résultats

- Par ordre croissant (ASC) ou décroissant (DESC).

- Syntaxe :

- ORDER BY expression [ASC | DESC]

- Exemples

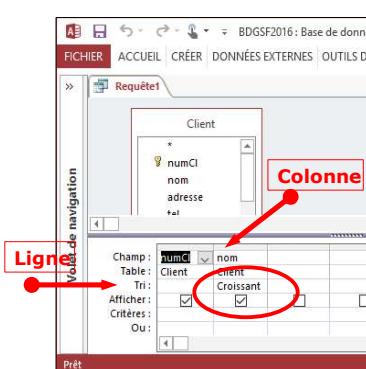
- Exemple 1 : Donner la liste des noms de clients par ordre alphabétique croissant.

**SELECT numCl, nom
FROM Client
ORDER BY nom ;**

ou
**SELECT numCl, nom
FROM Client
ORDER BY nom ASC;**

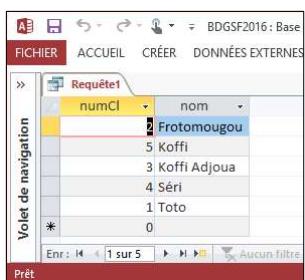
- Créer une nouvelle requête basée sur la table Client

- A l'intersection de la ligne "Tri" et la colonne "nom" sélectionner "Croissant"



- Exécuter la requête.

VII. Les requêtes



- Visualiser le code SQL générée.

```
SELECT Client.numCl, Client.nom  
FROM Client  
ORDER BY Client.nom;
```

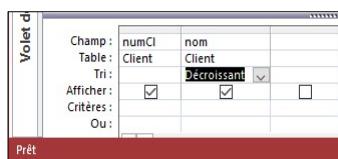
- Enregistrer la requête : RListeClientTri1
- Fermer la requête

VII. Les requêtes

- Exemple 2 : Donner la liste des noms de clients par ordre alphabétique décroissant.

```
SELECT numCl, nom  
FROM Client  
ORDER BY nom DESC ;
```

- Créer une nouvelle requête basée sur la table Client
- A l'intersection de la ligne "Tri" et la colonne "nom" sélectionner "Décroissant"



- Exécuter la requête



- Visualiser le code SQL générée

```
SELECT Client.numCl, Client.nom  
FROM Client  
ORDER BY Client.nom DESC;
```

- Enregistrer la requête : RListeClientTri2
- Fermer la requête

VII. Les requêtes

□ 7.5. Requêtes action

- Permettre de mettre à jour des données ou de créer de nouvelles colonnes résultant d'opération de calculs sur les autres colonnes.

- Syntaxe

```
UPDATE table  
SET colonne = exp, ...  
[ WHERE condition ]
```

- Requête 1 : On veut ajouter +8 à la quantité en stock 10 du produit sardine.

- Afficher la quantité en stock

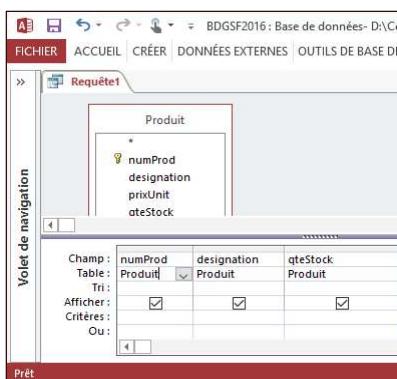
```
SELECT qteStock  
FROM Produit  
WHERE designation = 'Sardine';
```

- Requête de mise à jour

```
UPDATE Produit  
SET qteStock = qteStock + 8  
WHERE designation = 'Sardine';
```

VII. Les requêtes

- Créer une nouvelle requête basée sur la table Produit
- Ajouter les champs : numProd, designation, qteStock

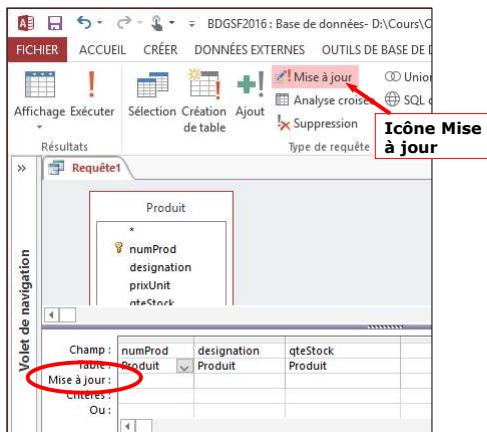


- Exécuter la requête pour voir les quantité en stock avant la mise à jour

Requête1				
	numProd	designation	prixUnit	qteStock
1	Savon		300	15
2	Eau Awa		100	29
3	Sardine		450	10
4	Assiette		245	27
5	Yaourt		250	50
*	0		0	0

- Dérouler la liste "Affichage" et sélectionner "Mode création"
- Cliquer sur l'icône "Mise à jour"

VII. Les requêtes



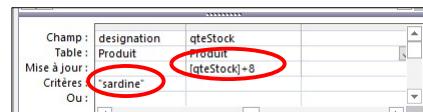
- Une nouvelle ligne "Mise à jour" a été ajoutée

INP-HBI/K. M. BROU

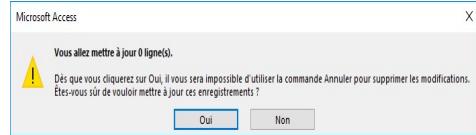
Chapitre 1 : Access niveau 1

161

- Saisir [qteStock]+8 à l'intersection de la ligne "Mise à jour" et de la colonne qteStock
- Saisir "Sardine" dans la zone Critère du champ designation



- Exécuter la requête



- Cliquer sur le bouton Oui

VII. Les requêtes

- Vérifier le résultat dans la table Produit

numProd	designation	prixUnit	qteStock
1	Savon	300	23
2	Eau Awa	400	37
3	Sardine	450	18
4	Assiette	245	35
6	Yaourt	250	58
*	0	0	0

- Visualiser le code SQL généré

```
UPDATE Produit SET Produit.qteStock = [qteStock]+8
WHERE (((Produit.designation)="sardine"));
```

- Remarque : l'exécution de la requête ne marche pas

- quantité en stock est 18 = 10 + 8
- Enregistrer la requête sous le nom RMajQteStockProduit

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

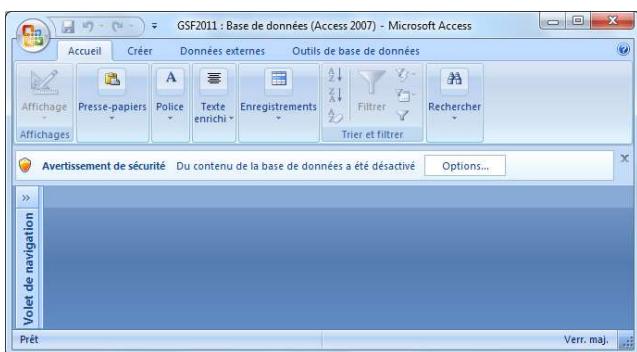
162

162

VII. Les requêtes

■ Remarque : l'exécution de la requête ne marche pas

- Cliquer sur Option
- Activer le contenu



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

163

163

VII. Les requêtes

■ Requête 2 : Insertion de données

□ Exemple : Ajouter un nouveau produit

```
INSERT INTO Produit VALUES(6, 'DVD', 400, 50);
```

■ Requête 3 : suppression de données

□ Exemple : supprimer le produit de numéro 6

```
INSERT INTO Produit VALUES(6, 'DVD', 400, 50);
```

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

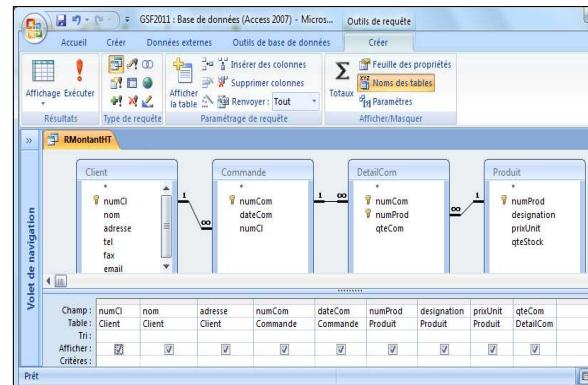
164

164

VII. Les requêtes

■ Requête 4 : Calculs des montants hors taxe.

- Ouvrir la requête RCommandeClient en mode création
- Enregistrer la sous le nom RMontantHT



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

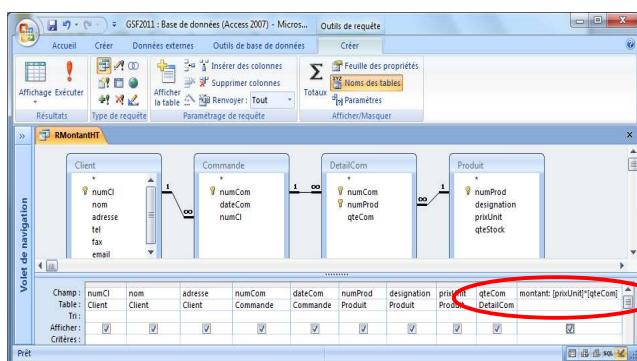
165

165

VII. Les requêtes

■ Cliquer dans le champ vide après PrixUnit et taper l'expression suivante :

▪ montant : qteCom * prixUnit



▪ Exécuter la requête

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

	numCl	nom	adresse	numCom	dateCom	numProd	designation	prixUnit	qteCom	montant
1	Toto	Yakro	1	13/11/2003	1 savon	300	10	3000		
1	Toto	Yakro	1	13/11/2003	2 Eau Awa	400	4	1600		
1	Toto	Yakro	1	13/11/2003	3 sardine	350	2	700		
1	Toto	Yakro	2	12/05/2004	2 Eau Awa	400	3	1200		
1	Toto	Yakro	2	12/05/2004	5 Yaourt	250	4	1000		
1	Toto	Yakro	3	13/02/2003	1 savon	300	2	600		
1	Toto	Yakro	3	13/02/2003	2 Eau Awa	400	6	2400		
1	Toto	Yakro	3	13/02/2003	3 sardine	350	3	1050		
2	Frotomougou	Abidjan	4	15/05/2004	2 Eau Awa	400	7	2800		
2	Frotomougou	Abidjan	4	15/05/2004	1 savon	300	6	1800		
2	Frotomougou	Abidjan	4	15/05/2004	5 Yaourt	250	5	1250		

166

166

VII. Les requêtes

□ Visualiser le code SQL générée

```
SELECT Client.numCl, Client.nom, Client.adresse,
Commande.numCom, Commande.dateCom, Produit.numProd,
Produit.designation, Produit.prixUnit, DetailCom.qteCom,
[prixUnit]*[qteCom] AS montant
FROM Produit INNER JOIN ((Client INNER JOIN Commande ON
Client.numCl = Commande.numCl) INNER JOIN DetailCom ON
Commande.numCom = DetailCom.numCom) ON Produit.numProd
= DetailCom.numProd;
```

□ Montant total de la commande numéro 1

```
SELECT cl.numCl, nom, co.numCom, dateCom, sum(qteCom *
prixUnit) AS Montant
FROM Client cl, Commande co, DetailCom de, Produit pr
WHERE cl.numCl = co.numCl AND
co.numCom = de.numCom AND
de.numProd = pr.numProd AND
co.numCom = 1
GROUP BY cl.numCl, nom, co.numCom, dateCom;
```

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

167

167

VII. Les requêtes

■ Autres Exemples

□ Calcul des montants de chaque produit commandé

```
SELECT Client.numCl, Client.nom, Commande.numCom,
Commande.dateCom, Produit.numProd,
Produit.designation,
Produit.prixUnit, DetailCom.qteCom, qteCom * prixUnit AS
Montant
FROM Produit
INNER JOIN ((Client INNER JOIN Commande ON
Client.numCl = Commande.numCl)
INNER JOIN DetailCom ON Commande.numCom =
DetailCom.numCom
)
ON Produit.numProd = DetailCom.numProd;
```

Ou

```
SELECT cl.numCl, nom, co.numCom, dateCom,
pr.numProd, designation, prixUnit, qteCom, qteCom * prixUnit
AS Montant
FROM Produit pr
INNER JOIN ((Client cl INNER JOIN Commande co ON cl.numCl
= co.numCl)
INNER JOIN DetailCom de ON co.numCom = de.numCom
)
ON pr.numProd = de.numProd;
```

numCl	nom	numCom	dateCom	numProd	designation	prixUnit	qteCom	Montant
1 Toto Ali		1 13/11/2003		1 savon	300	10	3000	
1 Toto Ali		1 13/11/2003		2 Eau Awa	400	4	1600	
1 Toto Ali		2 12/05/2004		2 Eau Awa	400	3	1200	
1 Toto Ali		3 13/02/2003		1 savon	300	2	600	
1 Toto Ali		3 13/02/2003		2 Eau Awa	400	6	2400	
1 Toto Ali		3 13/02/2003		3 sardine	450	3	1350	
2 Frotomogou		4 15/05/2004		2 Eau Awa	400	7	2800	
2 Frotomogou		5 18/05/2004		2 Eau Awa	400	3	1200	
2 Frotomogou		5 18/05/2004		3 sardine	450	6	2700	
3 Koffi Adjoua		6 12/03/2004		2 Eau Awa	400	1	400	
3 Koffi Adjoua		7 06/04/2003		3 sardine	450	5	2250	

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

168

168

VII. Les requêtes

□ Calcul du montant total de chaque commande

```
SELECT cl.numCl, nom, co.numCom, dateCom,
sum(qteCom * prixUnit) AS Montant
FROM Produit pr
INNER JOIN ((Client cl INNER JOIN Commande co ON
cl.numCl = co.numCl)
INNER JOIN DetailCom de ON co.numCom =
de.numCom
)
ON pr.numProd = de.numProd
GROUP BY co.numCom, cl.numCl, nom, dateCom;
```

numCl	nom	numCom	dateCom	Montant
1	Toto Ali	1	13/11/2003	4600
1	Toto Ali	2	12/05/2004	1200
1	Toto Ali	3	13/02/2003	4350
2	Frotomogou	4	15/05/2004	2800
2	Frotomogou	5	18/05/2004	3900
3	Koffi Adjoua	6	12/03/2004	400
3	Koffi Adjoua	7	05/04/2003	2250
3	Koffi Adjoua	8	06/05/2004	3200
4	Seri	9	06/05/2004	900
4	Seri	10	12/05/2004	1600

VII. Les requêtes

□ Calcul du montant total de la commande numéro 1

```
SELECT cl.numCl, nom, co.numCom, dateCom,
sum(qteCom * prixUnit) AS Montant
FROM Produit pr
INNER JOIN ((Client cl INNER JOIN Commande co ON
cl.numCl = co.numCl)
INNER JOIN DetailCom de ON co.numCom = de.numCom
)
ON pr.numProd = de.numProd
WHERE co.numCom = 1
GROUP BY co.numCom, cl.numCl, nom, dateCom;
```

numCl	nom	numCom	dateCom	Montant
1	Toto Ali	1	13/11/2003	4600

VII. Les requêtes

□ 7.4. Requêtes SQL spécifiques

■ Requête Union

- Combine des champs (colonnes) issus d'une ou plusieurs tables ou requêtes en un champ ou une colonne dans les résultats de la requête.
- Exemple : union de la table AncienProduit et nouveauProduit pour créer la table Produit.

■ Requête directS

- Envoie des commandes directement à des BD ODBC telles que Microsoft SQL Server, en utilisant des commandes reconnues par le serveur.
- Exemple : utiliser une requête directe pour récupérer des enregistrements ou modifier des données.

■ Requête Définition des données

- Crée ou modifie des objets BD, tels que des tables Microsoft Access ou Microsoft SQL Server.

VII. Les requêtes

■ Sous-requête

- Constituée d'une instruction SQL SELECT contenue dans une autre requête de sélection ou dans une requête action.
 - Instructions à mettre dans la ligne Champ de la grille de création de la requête afin de définir un nouveau champ ou dans la ligne Critères afin de définir les critères d'un champ.

□ Utiliser les sous-requêtes pour :

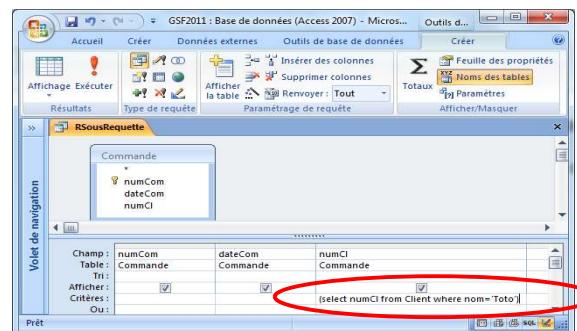
- Tester l'existence de certains résultats à partir de la sous-requête (en utilisant les mots réservés EXISTS ou NOT EXISTS).
- Rechercher les valeurs contenues dans la requête principale, qui sont égales, supérieures ou inférieures aux valeurs renvoyées par la sous-requête (en utilisant les mots réservés ANY, IN ou ALL).
- Créer des sous-requêtes dans les sous-requêtes (sous-requêtes imbriquées).

VII. Les requêtes

Exemple : Commandes du client de nom Toto

- Créer une nouvelle requête basée sur la table commande
- Ajouter tous les champs
- Saisir la sous-requête suivante dans la zone Critère du champ numCl :

```
=select numCl from Client where nom='Toto')
```



VII. Les requêtes

Exécuter la requête

numCom	dateCom	numCl
1	13/11/2003	1
2	12/05/2004	1
3	13/02/2003	1
*	0	0

Visualiser la requête

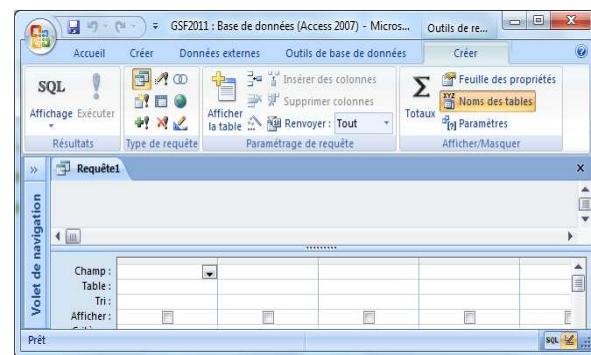
```
SELECT Commande.numCom, Commande.dateCom,
       Commande.numCl
  FROM Commande
 WHERE ((Commande.numCl)=(select numCl from
Client where nom='Toto'));
```

Enregistrer (RSousRequette) et fermer la requête

VII. Les requêtes

□ 7.5. Création de requêtes en mode SQL

- Crée une requête en utilisant directement les ordres SQL
- Exemple : liste des clients de Yakro
 - Crée une nouvelle requête
 - Fermer la fenêtre "Ajout de tables" sans ajouter une table



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

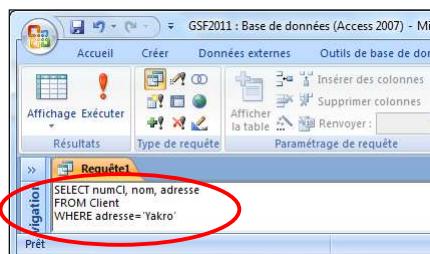
175

175

VII. Les requêtes

- Dérouler le menu Affichage et sélectionner "SQL Mode SQL"
- Saisir la requête suivante

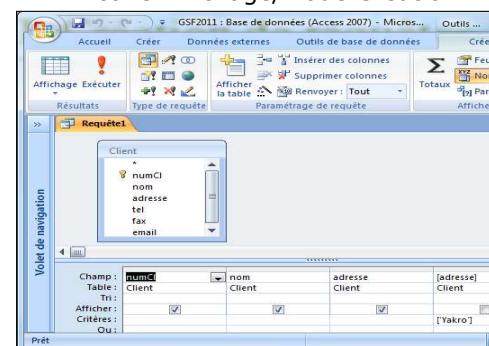
```
SELECT numCl, nom, adresse  
FROM Client  
WHERE adresse='Yakro';
```



- Exécuter la requête

□ Visualiser la requête graphique générée

- Icône Affichage/Mode Création



- Enregistrer (RListeClient5) et fermer la requête

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

176

176

VII. Les requêtes

■ Schéma relationnel avec le LDD d'ACCESS

- Chaque requête doit être exécutée et enregistrée dans un fichier différent.
- Créer une nouvelle BD appelée **GSF20112.accdb**
- Crée les requêtes suivantes :
 - Création de tables

Requête	Nom de la requête
CREATE TABLE Client2 (numCI LONG NOT NULL, nom TEXT(20), adresse TEXT(30), tel LONG, fax LONG, email TEXT(20));	RTClient2
CREATE TABLE Produit2 (numProd LONG NOT NULL, designation TEXT(15), prixUnit REAL, qteSock SHORT);	RTProduit2
CREATE TABLE Commande2 (numCom LONG NOT NULL, dateCom DATE, numCI LONG);	RCommande2
CREATE TABLE DetailCom2 (numCom LONG NOT NULL, numProd LONG NOT NULL, qteCom SHORT);	RDetailCom2

VII. Les requêtes

- Contraintes d'intégrités : clés primaires et clés étrangères

Requête	Nom de la requête
ALTER TABLE Client2 ADD CONSTRAINT Clients2_PK PRIMARY KEY (numCI);	RCIClient2PK
ALTER TABLE Produit2 ADD CONSTRAINT Produit2_PK PRIMARY KEY (numProd);	RCIProduit2PK
ALTER TABLE Commande2 ADD CONSTRAINT Commande2_PK PRIMARY KEY (numCom);	RCICommande2PK
ALTER TABLE DetailCom2 ADD CONSTRAINT DetailCom2_PK PRIMARY KEY (numCom, numProd);	RCIDetailCom2PK
ALTER TABLE Commande2 ADD CONSTRAINT Commande2_Client2_FK FOREIGN KEY (numCI) REFERENCES Client2 (numCI);	RCIComClient2FK
ALTER TABLE DetailCom2 ADD CONSTRAINT DetailCom2_Commande2_FK FOREIGN KEY (numCom) REFERENCES Commande2 (numCom);	RCIDetailComComFK
ALTER TABLE DetailCom2 ADD CONSTRAINT DetailCom2_Produit2_FK FOREIGN KEY (numProd) REFERENCES Produit2 (numProd);	RCIDetailComProdFK

VII. Les requêtes

- Insertion dans les tables

Requête	Nom de la requête
INSERT INTO Client2 VALUES(1, 'Toto', 'Yakro', 30642020, 30642020, 'toto@yahoo.fr');	RI1Client2
INSERT INTO Client2 VALUES(2, 'Frotomougou', 'Abidjan', 22343434, 22343434, "");	RI2Client2

VIII. Les formulaires

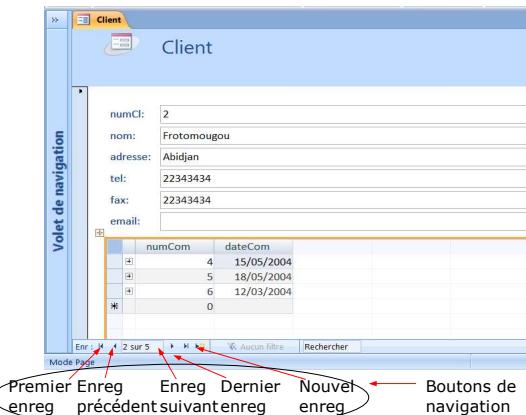
□ 8.1. Présentation

- Permettent de rendre plus agréables la présentation et la saisie des données dans les tables.
- Trois façons de créer les formulaires :
 - Création automatique
 - Création à l'aide de l'Assistant formulaire.
 - Création manuelle

VIII. Les formulaires

□ 8.2. Création automatique de formulaires

- Access crée un formulaire avec un format prédéfini.
- Cliquer la table Clients
- Cliquer sur l'onglet Créer
- Cliquer sur l'icône "Formulaire"



□ Enregistrer et fermer le formulaire

INP-HBI/K. M. BROU

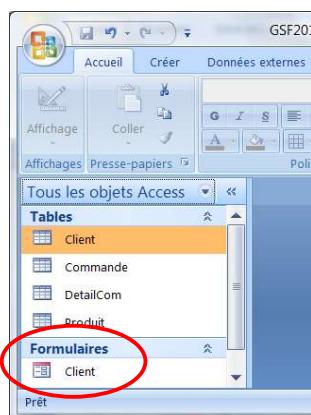
Chapitre 1 : Access niveau 1

181

181

VIII. Les formulaires

- Dérouler la liste "Tous les objets d'Access" et sélection "Type d'objet"



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

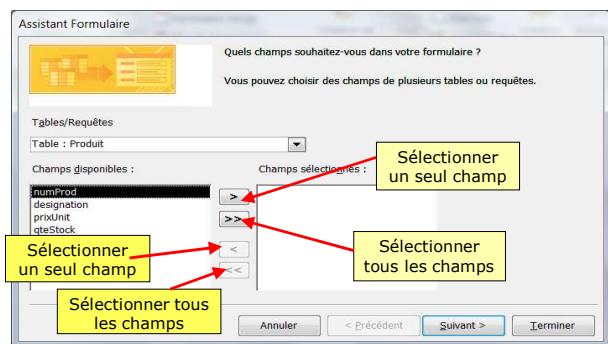
182

182

VIII. Les formulaires

□ 8.3. Assistant formulaire

- Guide étape par étape dans la création du formulaire.
- Sélectionner l'objet Créer
- Cliquer sur le bouton "Plus de formulaires"
- Sélectionner "Assistant Formulaire"



INP-HBI/K. M. BROU

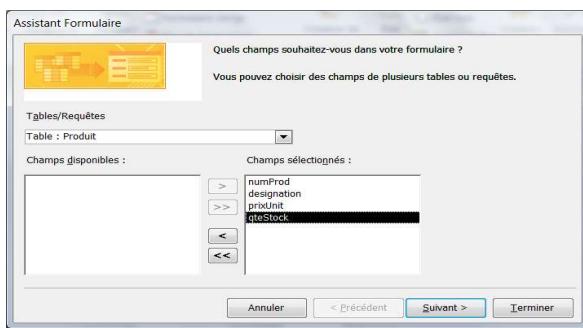
Chapitre 1 : Access niveau 1

183

183

VIII. Les formulaires

- Sélectionner la table Produit dans la liste déroulante
- Sélectionner les champs qui doivent apparaître dans le formulaire



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

- Cliquer sur le bouton Suivant
- choisir une disposition et cliquer sur le bouton Suivant.

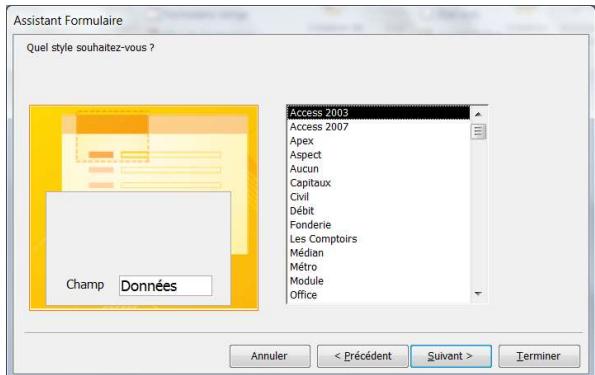


- Choisir une disposition et cliquer sur le bouton Suivant.

184

184

VIII. Les formulaires



- **Tester les différentes dispositions**
- **Choisir la disposition "Access 2003" et cliquer sur le bouton Suivant.**
 - Certaines dispositions sont difficiles à mettre à jour.

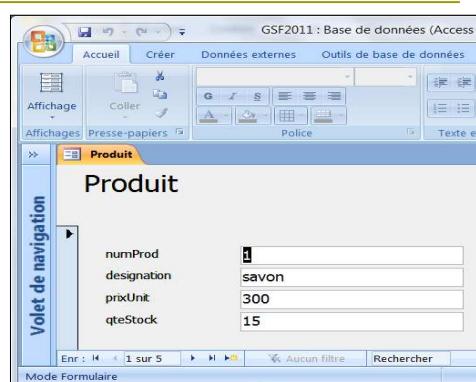
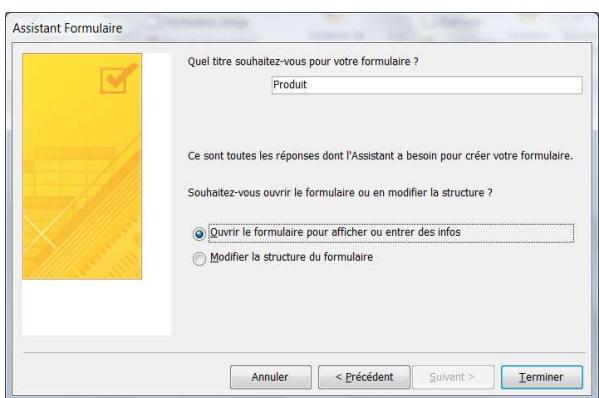
INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

185

185

VIII. Les formulaires



- **Cliquer sur le bouton Terminer.**
- **Enregistrer le formulaire (sous le nom Produit)**
- **Ajouter au moins 2 nouveaux produits.**

INP-HBI/K. M. BROU

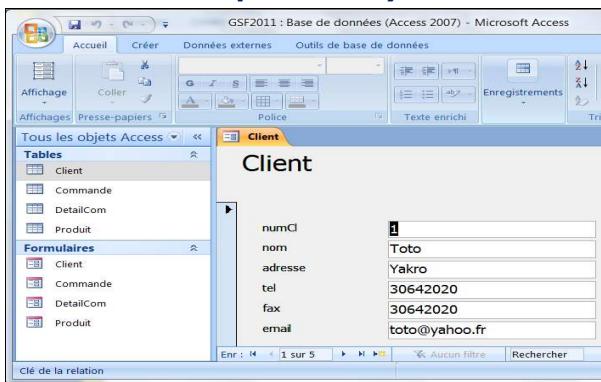
Chapitre 1 : Access niveau 1

186

186

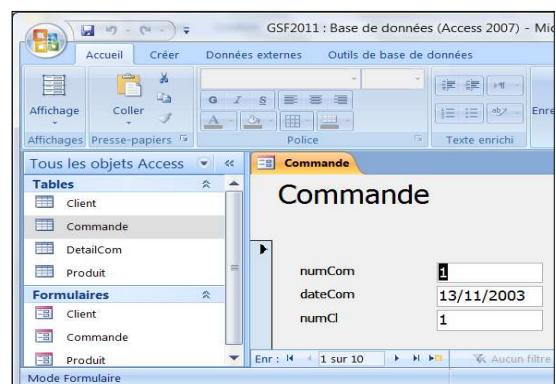
VIII. Les formulaires

- Procéder de même pour créer le formulaire associé à la table Client (après avoir supprimer le formulaire précédent).



Chapitre 1 : Access niveau 1

- Procéder de même pour créer le formulaire associé à la table Commande.

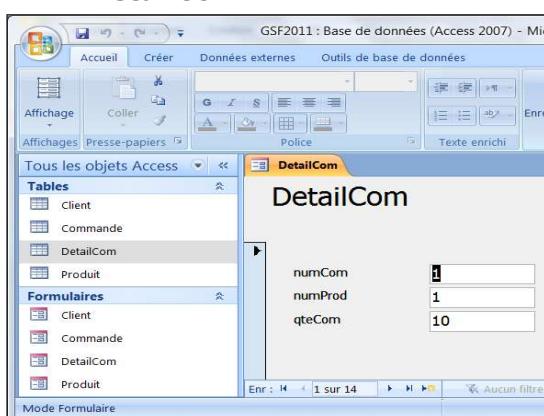


187

187

VIII. Les formulaires

- Procéder de même pour créer le formulaire associé à la table DetailCom.



Chapitre 1 : Access niveau 1

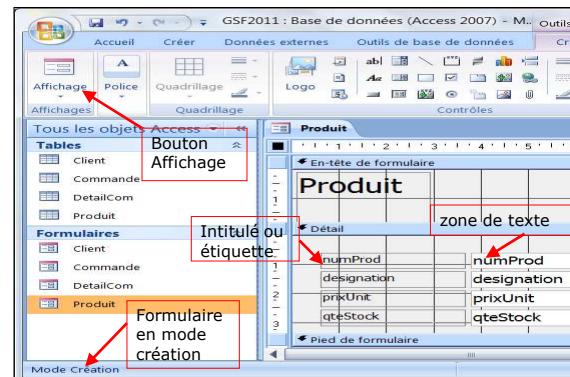
188

188

VIII. Les formulaires

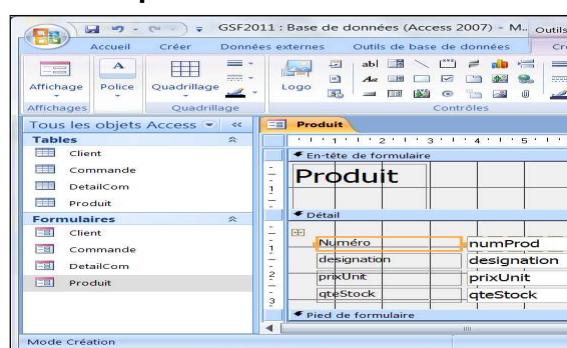
□ 8.4. Mise en forme

- Ouvrir le formulaire Produit, puis dérouler la liste de l'icône Affichage/Mode création
- Ou Clic droit sur le formulaire Produit/Mode création

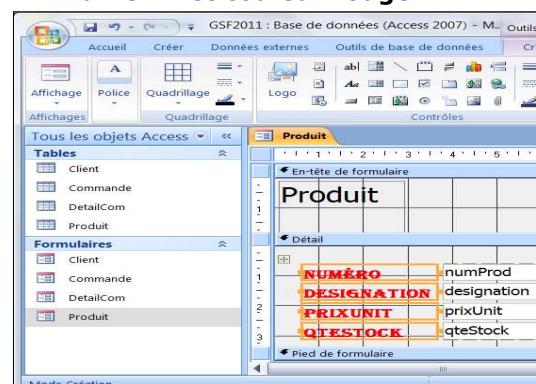


VIII. Les formulaires

- Mise en forme des étiquettes et zones de saisie
 - Cliquer deux fois sur l'étiquette numProd et remplacer le contenu par Numéro.



- Sélectionner toutes les étiquettes.
- Choisir la police Algerian, Gras, Taille 12 et couleur rouge



VIII. Les formulaires

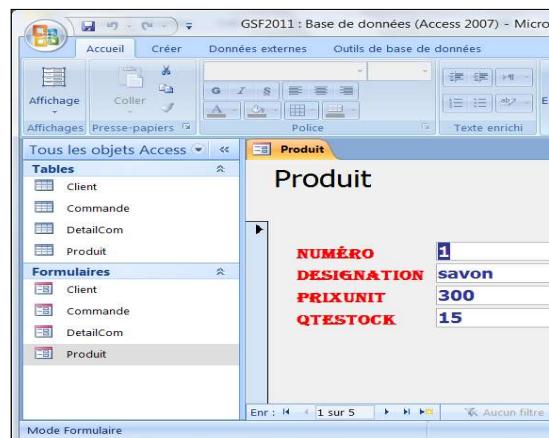
- Sélectionner toutes les zones de saisie.
- Choisir la police Algerian, Gras, Taille 12 et couleur bleue



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

- Cliquer sur l'icône Affichage pour apprécier le rendu

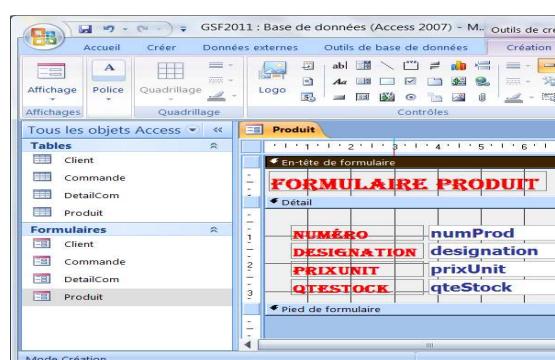


191

191

VIII. Les formulaires

- Mise en forme des entêtes et pieds de formulaire
- Cliquer deux fois dans l'étiquette de l'entête de formulaire et saisir : Formulaire Produit.
- Choisir la police Algerian, Gras, Taille 16 et couleur rouge



INP-HBI/K. M. BROU

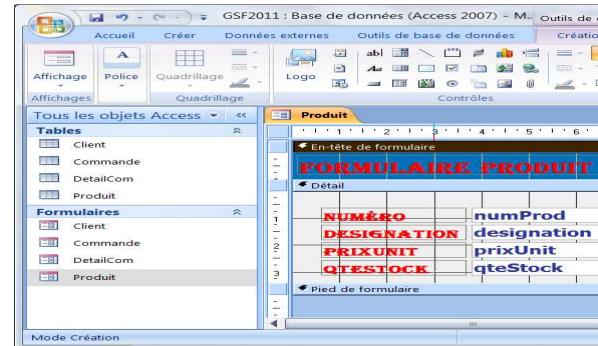
Chapitre 1 : Access niveau 1

192

192

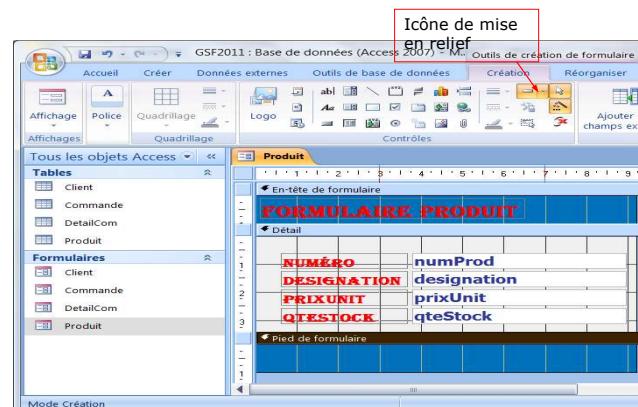
VIII. Les formulaires

- Réduire l'espace entre la barre En-tête de formulaire et Détail d'environ 1 cm.
- Cliquer dans l'en-tête de formulaire (en dehors de l'étiquette produit).
- Cliquer sur l'icône Couleur d'arrière plan et sélectionner la couleur bleue.



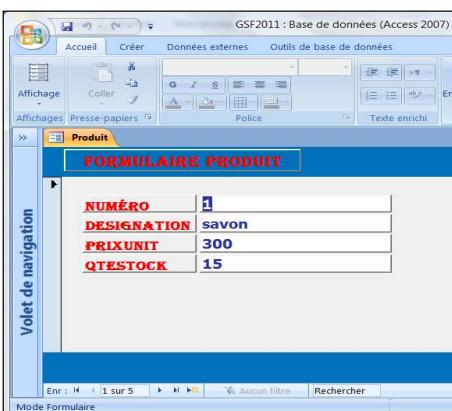
VIII. Les formulaires

- Agrandir le pied de formulaire d'environ 1 cm.
- Cliquer dans le pied de formulaire et sélectionner la couleur arrière plan bleue.
- Mettre en relief tous les contrôles



VIII. Les formulaires

■ Tester le formulaire



■ Faire la mise forme pour les autres formulaires

■ Enregistrer et fermer le formulaire Produit.

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

195

195

VIII. Les formulaires

■ Formulaire Client



■ Faire formulaire Commande



INP-HBI/K. M. BROU

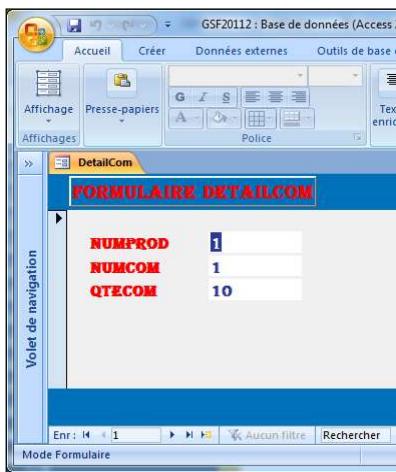
Chapitre 1 : Access niveau 1

196

196

VIII. Les formulaires

■ Faire formulaire DetailCom



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

197

197

VIII. Les formulaires

□ 8.5. Créer un menu général

- **Créer un formulaire qui présente toutes les fonctionnalités de l'application.**



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

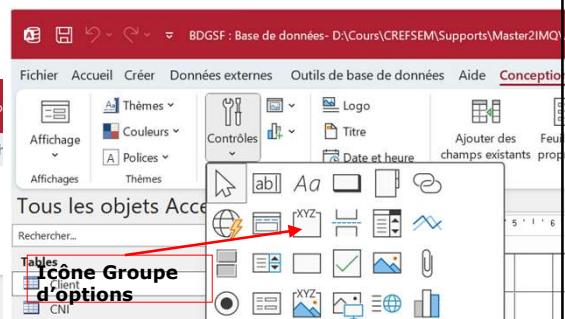
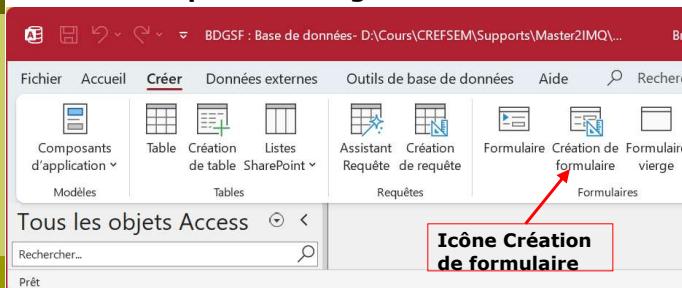
198

198

VIII. Les formulaires

■ Processus

□ Cliquer sur l'onglet Crée



□ Cliquer sur l'icône "Création de formulaire"

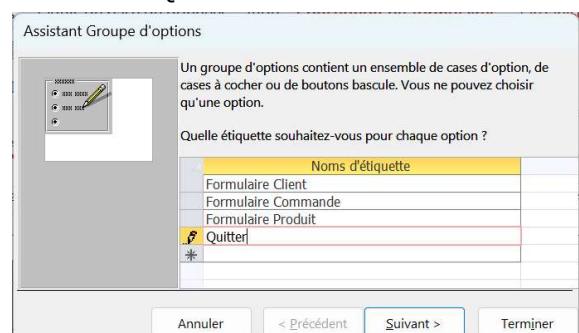
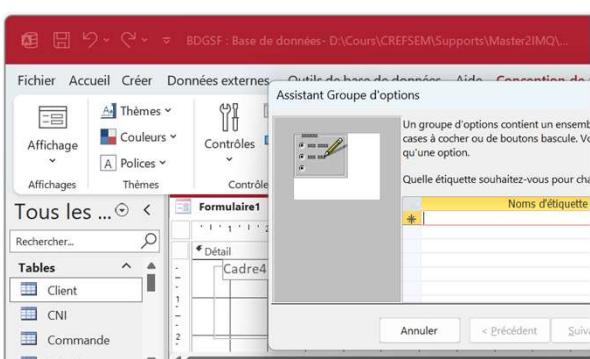
□ Dans le groupe de commande Contrôle, "Groupe d'options"

□ Tracer un rectangle d'environ 5 cm de largeur et 10 cm de hauteur dans le formulaire

VIII. Les formulaires

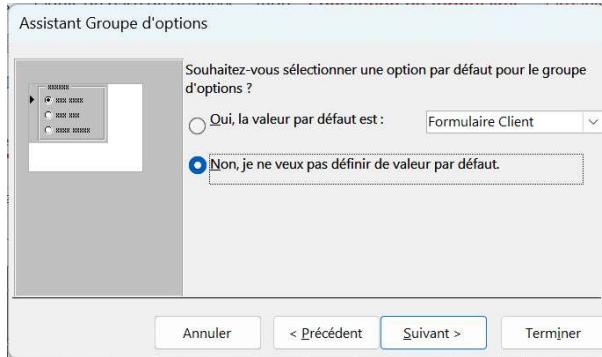
□ Saisir les étiquettes suivantes :

- Formulaire Client, Formulaire Produit, Formulaire Commande et Quitter

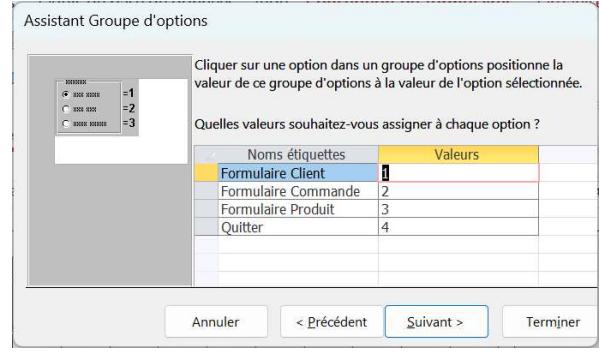


□ Cliquer sur le bouton Suivant

VIII. Les formulaires



- Sélectionner la deuxième option et cliquer sur le bouton Suivant



- Cliquer sur le bouton Suivant

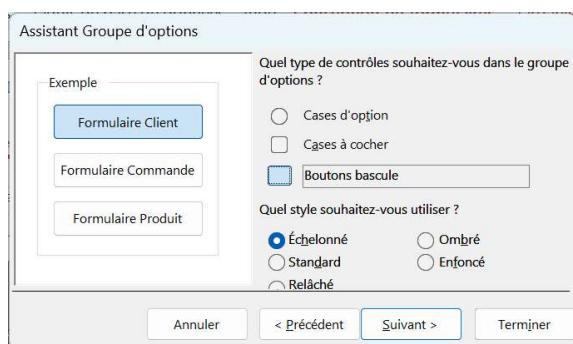
INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

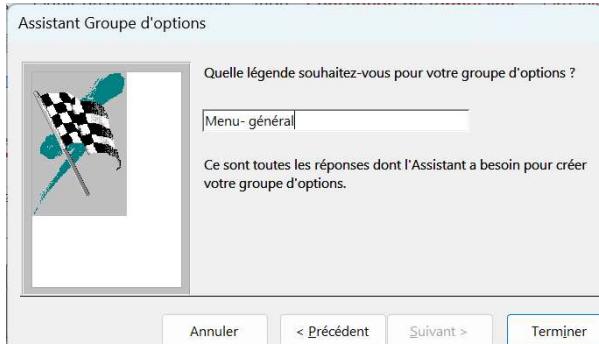
201

201

VIII. Les formulaires



- Sélectionner le contrôle Bouton bascule et cliquer sur le bouton Suivant



- Saisir la légende : Menu général
- Cliquer sur le bouton Terminer

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

202

202

101

VIII. Les formulaires

Formulaire1

Formulaire Client

Formulaire Commande

Formulaire Produit

Quitter

Formulaire1

Formulaire Client

Formulaire Commande

Formulaire Produit

Quitter

Faire la mise en forme

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

203

203

VIII. Les formulaires

Formulaire1

Formulaire Client

Formulaire Commande

Formulaire Produit

Quitter

Tester le formulaire

Cliquer sur les différentes boutons

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

Pour le moment les boutons ne fonctionnent pas.

Enregistrer le formulaire sous le nom : menuGeneral et fermer le.

204

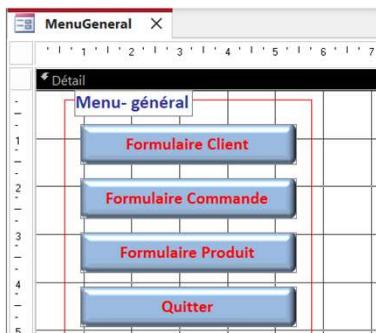
204

102

VIII. Les formulaires

■ Lier les macros au formulaire MenuGeneral

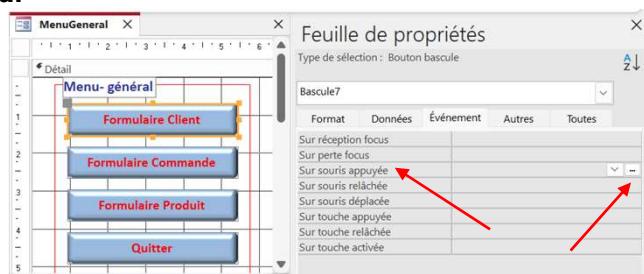
- Ouvrir le formulaire Menu général en mode création



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

Feuille de propriétés



- Cliquer sur l'onglet Événement
- Cliquer sur le bouton ... de l'événement "Sur souris appuyée"

205

205

VIII. Les formulaires

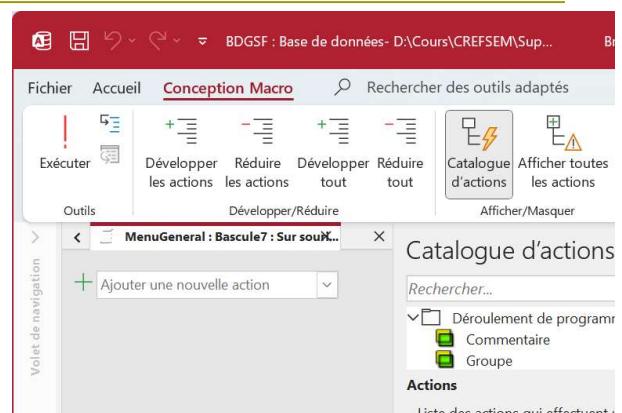
- Sélectionner "Générateur de macros" et cliquer sur OK



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

- Dérouler la liste des actions et sélectionner- la macro "OuvrirFormulaire"



206

206

VIII. Les formulaires

The screenshot shows the Microsoft Access ribbon at the top. Below it, there are two windows: a 'Macro' window titled 'MenuGeneral : Bascule7 : Sur souris app...', and a 'Feuille de propriétés' (Properties Sheet) window.

Macro Window: This window contains a single step named 'Bascule7'. It has a 'Type de sélection' (Selection type) of 'Bouton bascule' (Check button). The 'Événement' (Event) is set to 'Sur souris appuyée' (On mouse down). The 'Format' tab is selected. The macro itself consists of a single action: 'OuvrirFormulaire' (OpenForm) with the following parameters:

- Nom de formulaire: Client
- Affichage: Formulaire
- Nom de filtre: (empty)
- Condition Where: (=)
- Mode Données: (empty)
- Mode Fenêtre: Standard

Properties Sheet: This window shows the properties for the 'Bascule7' macro. The 'Format' tab is selected. The events listed are:

- Sur réception focus
- Sur perte focus
- Sur souris appuyée [Macro incorporée]
- Sur souris relâchée
- Sur souris déplacée
- Sur touche appuyée
- Sur touche relâchée
- Sur touche activée

Task List: A list of tasks is displayed below the windows:

- Sélectionner le formulaire Client
- Fermer la fenêtre des macros
- Ouvrir le formulaire en mode formulaire

INP-HBI/K. M. BROU Chapitre 1 : Access niveau 1 207

207

VIII. Les formulaires

The screenshot shows the Microsoft Access ribbon at the top. Below it, there are two windows: a 'Macro' window titled 'MenuGeneral' and a 'Form' window titled 'Client'.

Macro Window: This window shows a menu macro named 'Menu- général'. It contains four items: 'Formulaire Client', 'Formulaire Commande', 'Formulaire Produit', and 'Quitter'. The 'Formulaire Client' item is highlighted with a red border.

Form Window: This window displays a 'Client' form with the following fields and data:

numéro	1
nom	Toto Ali
adresse	Yakro
tel	30642020
fax	306420200
email	toto@yahoo.f
numCld	C001

Below the form, a subgrid shows a list of commands:

numCom	dateCom
1	13/11/2003
2	12/05/2004
3	13/02/2003
*	0

Task List: A list of tasks is displayed below the windows:

- Cliquer sur le bouton Formulaire Client

INP-HBI/K. M. BROU Chapitre 1 : Access niveau 1 208

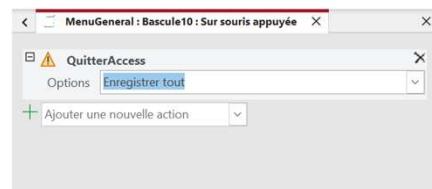
208

VIII. Les formulaires

■ Exercice

- Faire de même pour les autres boutons sauf le bouton Quitter

- Pour le bouton Quitter, sélectionner la macro QuitterAccess



- Ouvrir le formulaire en mode formulaire pour tester le bouton Quitter

IX. Les Etats

□ 9.1. Présentation

- Un état permet de représenter les données sous une forme imprimée.

- Définir la taille et la mise en forme de chaque élément de l'état.
- Afficher les informations exactement comme on le souhaite.

- Source de données

- La plupart des informations d'un état proviennent de tables sous-jacentes, de requêtes ou d'instructions SQL constituant la source des données de l'état.

- Les autres informations contenues dans l'état sont, quant à elles, enregistrées lors de sa création.

■ Création d'un état

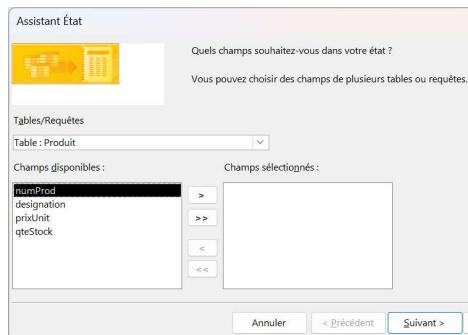
- On crée le lien entre un état et ses enregistrements source en utilisant les contrôles de la Boîte à outils

IX. Les Etats

□ 9.2. Liste des clients

■ Création de l'état

- Sélectionnez l'onglet Créer
- Cliquez sur le bouton "Assistant état"

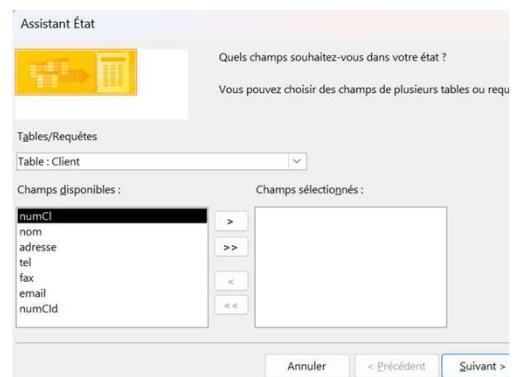


INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

211

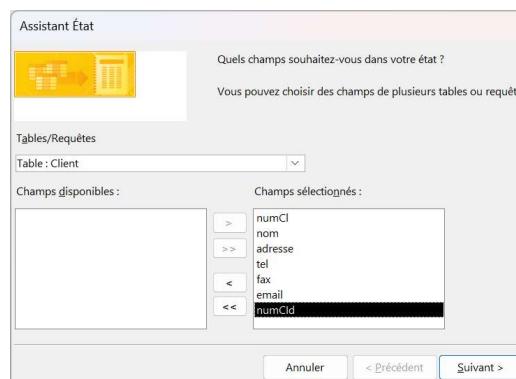
□ Sélectionnez la table Client



211

IX. Les Etats

- Sélectionnez tous les champs et cliquez sur le bouton Suivant



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

212



- Enlever le niveau de regroupement existant

212

IX. Les Etats

□ Sélectionner le niveau de regroupement par adresse

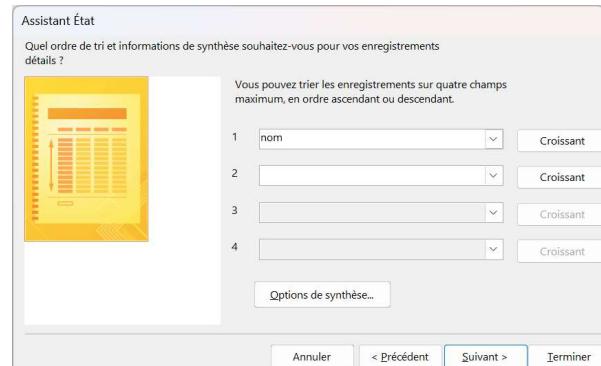


□ Cliquer sur le bouton Suivant

INP-HBI/K. M. BROU

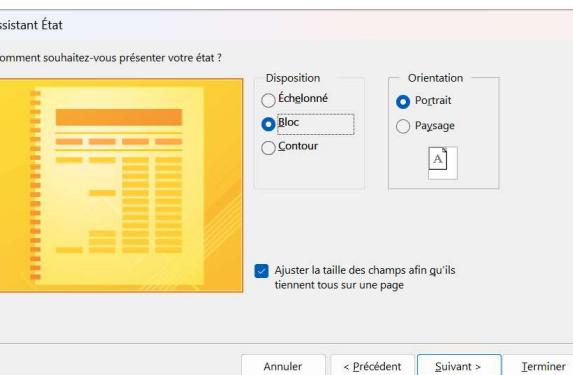
Chapitre 1 : Access niveau 1

213



□ Triez les enregistrements par nom, et cliquer sur le bouton Suivant

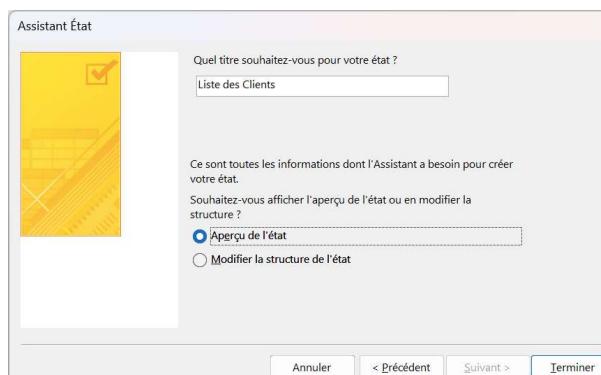
IX. Les Etats



□ Choisissez une Disposition, une orientation et cliquez sur le bouton Suivant

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1



□ Donner un titre (Liste des clients) et cliquer sur le bouton Terminer

214

IX. Les Etats

The screenshot shows a Microsoft Access window titled "Liste des Clients". The grid displays the following data:

adresse	nom	numCl	tel	fax	email	num
Abidjan	Frotomougou	2	22343434	22343434		C00
Bouaké	Koffi Adjoua	3	30636565	30636565		C00
	Kouakou	45	31	31	kouakou	C00
Daloa	Séri	4	34343434	34343434	seri@gmail.com	C00
Tipatipa	Kouadio	145	30	30	kouadio	C00
Yakro	Koffi	5	0	0		C00
	Toto	1	30642020	30642020	toto@yahoo.fr	C00

215

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

215

IX. Les Etats

The screenshot shows a Microsoft Access window titled "Liste des Clients". The grid displays the following data:

Adresse	Nom	NumCl	Tel	Fax	Email	NumCl
Abidjan	Frotomougou	2	22343434	22343434		C00;
Bouaké	Koffi Adjoua	3	30636565	30636565		C00;
	Kouakou	45	31	31	kouakou	C00I
Daloa	Séri	4	34343434	34343434	seri@gmail.com	C00;
Tipatipa	Kouadio	145	30	30	kouadio	C00;
Yakro	Koffi	5	0	0		C00;
	Toto	1	30642020	30642020	toto@yahoo.fr	C00;

Annotations in red text at the bottom left:

- lundi 9 septembre 2024
- Page 1 sur 1
- Faire la mise en forme de l'état

216

INP-HBI/K. M. BROU

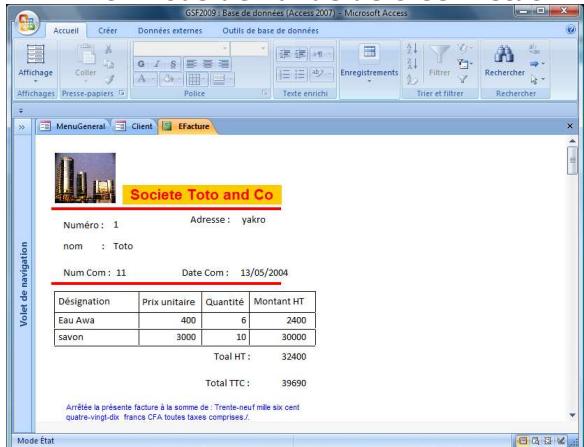
Chapitre 1 : Access niveau 1

216

IX. Les Etats

■ Exercice

□ On vous demande de créer l'état



INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

217

217

X. Projet

□ TP noté BD GestNotes

- Soient les tables suivantes et leurs extensions :
 - Classe(codeCl, nomCl, effectif)
 - Etudiant(matricule, nomE, adresse, #codeCl)
 - Matiere(codeMa, nomM, coeff)
 - Composer(#matricule, #codeMa, note)

Etudiant			
matricule	nomE	adresse	codeCl
ESI01	TOTO	Yakro	INGINFO1
ESI02	FROTO	Yakro	INGINFO1
ESI03	MANKOU	Abidjan	INGINFO2

Classe		
codeCl	nom Cl	effectif
INGINFO1	INFO1	15
INGINFO2	INFO2	20

Matiere		
codeMa	nom M	coeff
M1	BD	2
M2	ALGO	1

Composer		
matricule	codeMa	notes
ESI01	M1	12
ESI01	M2	16
ESI02	M1	13
ESI02	M2	12
ESI03	M1	14
ESI03	M2	11

■ Créer la BD ci-dessus

INP-HBI/K. M. BROU

Chapitre 1 : Access niveau 1

218

218

X. Projet

■ Les requêtes

1. Liste des Etudiant.
2. Liste des Etudiant (matricule, nomE, adresse) ordonnée par ordre alphabétique croissant sur nomE.
3. Liste des Etudiant (matricule, nomE, adresse) de la classe de code INGINFO1.
4. Liste des Etudiant (matricule, nomE, adresse) de la classe de nom INFO1.
5. Liste des étudiant regroupés par classes (matricule, nomE, codeCl, nomCl).
6. Nombre d'étudiant par classe.
7. Moyennes générales de chaque étudiant de la classe INFO1 (matricule, nomE, moyenne).
8. Matricule et nom du premier de la classe INFO1.
9. Moyenne générale de la classe INFO1.
10. Moyenne générale de toutes les classes.

X. Projet

11. Créer l'état suivant :

CALCUL DE MOYENNE					
Année scolaire : 2023-2024			Semestre : 1		
Nom	BD	Math	IA	Moyenne	Rang
Fatou	10	11	9	10,00	3
Froto	15	6	5	9,57	4
Mankou	15	13	16	14,71	1
Toto	12	12	15	12,86	2

Coefficients	3	2	2
		Moyenne	11,79
		Moyenne Min	9,57
		Moyenne Max	14,71

Bibliographie

□ Livre

- "Modélisation dans la conception des systèmes d'information", Edition Masson.
- "Les fichiers et organisation des données", C. JOUFFROY, C.LEITANG, Bordas Informatique.
- "Base de données et systèmes relationnels", C. DELOBEL, M. ADIBA, Dunod Informatique.
- "Les bases de données relationnelles", Serge MIRANDA, José Maria BUSTA.
- "Base de données, les systèmes et leurs langages", G. GARDARIN, Edition Eyrolles.
- "Système d'information et base de données", GALACSI, Bordas Informatique.

Bibliographie

□ Webogaphie

- <http://www.infres.enst.fr/~dom/bd/polyv7/>
- <http://lbdwww.epfl.ch/f/teaching/courses/poly2/11/11.htm>
- Cours de Yolaine.Bourda@supelec.fr
- Cours de Mme Silber cours@www-aius.u-strasbg.fr