

SEA → SR

Équivalence

- Modèle E/A ↔ • Modèle relationnel
 - Association, entité
 - Relation (Table)
 - Propriété
 - Attribut
 - Identifiant
 - Clé primaire

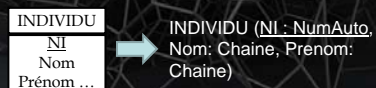
2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

42

SEA → SR

- Étape 1
 - Traduction des classes d'entités en Relations
 - Une entité ⇒ une relation
 - Un identifiant ⇒ la clé primaire
 - Une propriété ⇒ un attribut
 - Exemple



2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

43

SEA → SR

- Étape 2
 - Traduction des associations «?,n – ?,n» en Relations
 - Création d'une nouvelle relation R
 - Propriétés de l'association → attributs de R
 - Clé primaire de R est formée des identifiants des entités qui participent à la relation
 - Exemple



2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

44

SEA → SR

- Étape 3
 - Traduction des associations «?,1-?,?» en attributs
 - Identifiant de l'entité associé à «?,n» est ajouté comme clé étrangère dans la relation de l'entité associée à «?,1»
 - Exemple



→ FILM (NF: NumAuto, Titre : Chaîne, Année : Entier, #NumR : Entier)

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

45

SEA → SR

- Étape 4
 - Optimisation
 - Utilisation de règles pour optimiser les relations créées
 - Notion de normalisation
 - Non détaillé

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

46

SEA → SR

- Conclusion
 - Cahier des charges
 - Modèle E/A
 - Schéma relationnel (sans domaines)
 - INDIVIDU (NI, Nom, Prenom)
 - FILM(NF, Titre, Année, NumR#)
 - CINEMA(NC, Nom, Adresse)
 - PROJECTION(NF#, NC#, DateProj)
 - JOUE(NI#, NF#, Role)

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

47

M114 – CM3 SQL – LDD

Denis PALLEZ
<http://denispallez.i3s.unice.fr>

Structured Query Language

- Langage
 - permettant de dialoguer avec SGBD
 - Propre à chaque SGBD
 - ⇒ Prendre le temps de regarder la **documentation**
 - Utilisable en mode intégré en programmation
 - Une requête = une chaîne de caractères
 - Manipulation d'objets correspondant à des enregistrements ou tables (plus récent)

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

49

Structured Query Language

- Catégories
 - Langage de Définition de Données (LDD – DDL)
 - Permet de créer, modifier et supprimer des tables dans une BD
 - Langage de Manipulation de Données (LMD – DML)
 - Permet de sélectionner, insérer, modifier ou supprimer des données dans une table d'une BD relationnelle
 - Langage de protection d'accès (LCD – DCL)
 - Définition des permissions au niveau des utilisateurs d'une BD
 - DCL (Data Control Language)
 - Langage de définition de Transactions (LDT – TDL)
 - Définition de transactions qui correspondent à des tâches

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

50

Structured Query Language

- En général
 - Caractères accentués acceptés mais fortement déconseillés
 - Espace accepté mais fortement déconseillé le nom doit être mis entre [] : [Date de Naissance]
 - La casse n'est souvent pas respectée (cf. doc)
 - Certains mots sont réservés
 - Date, Time, Not, Create
- Utiliser un éditeur de texte (TextEdit, SC1...)

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

51

SQL – Code source

Devinez lequel est le bon ?

```
SELECT DISTINCT VILLE_ETP
FROM T_ENTREPOT WHERE
RAYON_RYN IN (SELECT
RAYON_RYN FROM T_ENTREPOT
WHERE RAYON_RYN NOT IN
(SELECT RAYON_RYN FROM
T_ENTREPOT WHERE
RAYON_RYN NOT IN (SELECT
RAYON_RYN FROM T_RAYON))
) GROUP BY VILLE_ETP
HAVING COUNT (*) =
(SELECT COUNT (*)
FROM T_ENTREPOT
WHERE RAYON_RYN IN (SELECT RAYON_RYN
FROM T_ENTREPOT
WHERE RAYON_RYN NOT IN
(SELECT RAYON_RYN NOT IN
(SELECT T_RAYON_RYN
FROM T_RAYON)))
GROUP BY VILLE_ETP
HAVING COUNT (*) = (SELECT COUNT(DISTINCT RAYON_RYN)
FROM T_RAYON)
```

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

52

BD Exemple

- CATEGORIE (IdCat: NumAuto, Libelle:Texte(10), TauxTva:Reel)
- PRODUIT (IdPro: NumAuto, IdCat#:Entier, Designation:Texte, Marque:Texte, PrixUht:Reel, Qstock:Entier)
- CLIENT (IdCli:NumAuto, Nom:Texte, Ville:Texte)
- VENTE (IdCli#:Entier, IdPro#:Entier, DateVente:Date, Qte:Entier)
- COMMERCIAL (Matricule:Texte, Nom:Texte, Prénom:Texte, Ville:Texte)

2015-2016 M114 – Intro. BD – <http://denispallez.i3s.unice.fr> 53

SQL – LDD

- CREATE TABLE
- ALTER TABLE
- DROP TABLE
- Souvent possible de le faire interactivement (Access, PhpMyAdmin...)
- Utile pour
 - Définition de scripts en programmation
 - Interopérabilité entre SGBD
 - Quand pas d'IHM (serveur web)

2015-2016 M114 – Intro. BD – <http://denispallez.i3s.unice.fr> 54

LDD – Create

- Objectif
Création de la structure d'une table
- Syntaxe


```
CREATE TABLE <table> (
  <attribut1> <type_attribut1> <contraintes_attribut1>,
  <attribut2> <type_attribut2> <contraintes_attribut2>,
  ...,
  <attributn> <type_attributn> <contraintes_attributn>
);
```
- Exemple


```
CREATE TABLE client (
  IdCli          COUNTER,
  Nom            CHAR(20) NOT NULL,
  Ville         CHAR(30),
  Adresse       VARCHAR(50)
);
```

2015-2016 M114 – Intro. BD – <http://denispallez.i3s.unice.fr> 55

LDD – Domaines

- Différents d'un SGBD à l'autre (cf. doc)

Type de donnée	Syntaxe	Description
Type alphanumérique	CHAR(n)	Chaîne de caractères de longueur fixe n (n<16383)
Type alphanumérique	VARCHAR(n)	Chaîne de caractères de longueur variable, à n caractères maximum (n<16383)
Type numérique	NUMBER(n, [d])	Nombre de n chiffres [optionnellement d après la virgule]
Type numérique	SMALLINT	Entier signé de 16 bits (-32768 à 32767)
Type numérique	INTEGER	Entier signé de 32 bits (-2E31 à 2E31-1)
Type numérique	FLOAT	Nombre à virgule flottante
Type horaire	DATE	Date sous la forme 16/07/99
Type horaire	TIME	Heure sous la forme 12:54:24.85
Type horaire	TIMESTAMP	Date et Heure
Type texte	TEXT	Texte (Chaîne de caractères) de longueur variable (n<65535)
Type blob	BLOB	Objet binaire de longueur variable (n<65535)
Type Auto	ACCESS : AUTOINCREMENT	Numérotation automatique

2015-2016 M114 – Intro. BD – <http://denispallez.i3s.unice.fr> 56

LDD – Contraintes d'intégrité

- Demande au SGBD de vérifier en permanence certaines contraintes pour assurer une cohérence des données
- Mots clés correspondants
 - CONSTRAINT : nomme une contrainte
 - DEFAULT : valeur par défaut
 - NOT NULL : champ obligatoire
 - UNIQUE : champ unique
 - CHECK : vérifie une contrainte particulière

Default

- Valeur par défaut
 - Utiliser le mot clé DEFAULT <valeur>
 - Facilite la création d'enregistrements et garantit que le champ n'est pas vide
 - Valeur :
 - constante numérique
 - constante alphanumérique (chaîne de caractères)
 - le mot clé USER (nom de l'utilisateur)
 - le mot clé NULL (par défaut)
 - le mot clé CURRENT_DATE (date de saisie)
 - le mot clé CURRENT_TIME (heure de saisie)
 - le mot clé CURRENT_TIMESTAMP (date et heure de saisie)

Contraintes

- Condition sur un champ
 - Utiliser le mot clé CHECK(<condition logique>)
 - Si valeur est différente de NULL, le SGBD vérifie que le champ satisfait la condition logique
 - Condition peut contenir des ordres SELECT...
- Unicité d'une valeur
 - La clause UNIQUE permet de vérifier que la valeur saisie n'existe pas déjà dans la table
 - Permet de garantir que toutes les valeurs d'un attribut sont différentes
- Nommer une contrainte
 - Utiliser le mot clé CONSTRAINT <nom_contrainte>
 - Lorsque la contrainte n'est pas respectée, le nom de la contrainte sera affichée par le SGBD
 - Des noms sont données par défaut par le SGBD mais sont incompréhensibles
- Clés Primaires (PRIMARY KEY)
 - Équivalent à UNIQUE + NOT NULL
- Clés étrangères (FOREIGN KEY ... REFERENCES ..)
 - Valeur doit exister dans une autre table

Exemple de contraintes

- Exemple de création avec contraintes d'intégrité

```
CREATE TABLE Clients (  
  IdCli Counter primary key,  
  Nom char(30) NOT NULL,  
  Prenom char(30) default 'Non renseigné',  
  Age integer check (age >0 and age < 150),  
  Email char(50) check (NOT NULL and Email LIKE "%@%")  
);
```

LDD – Particularités Access

- Access n'implémente pas toute la syntaxe SQL
- Types
 - Boolean
 - BIT ou YESNO (avec NON = 0, OUI = -1)
 - NumAuto
 - COUNTER, AUTOINCREMENT
 - Date
 - #20/07/2015# ou #2015/07/15#
 - Plus logique qu'une chaîne de caractères (autres SGBD)
- Pas de propriétés CHECK et DEFAULT

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

61

LDD – Alter

- Objectif
Modification (Altération) de la structure d'une table
- Syntaxe
`ALTER TABLE <table>
ADD <attribut> <type_attribut> <propriété_attribut>;`
- Exemples
 - Ajout du champ IDCat de type Numauto ds CATEGORIE
`ALTER TABLE Categorie
ADD IDcat COUNTER ;`
 - Le champ IDCat de CATEGORIE devient clé primaire
`ALTER TABLE Categorie
ADD CONSTRAINT pk_Categorie PRIMARY KEY (IDcat) ;`
 - Le champ IDCat devient clé étrangère ds PRODUIT et référence la clé primaire de CATEGORIE
`ALTER TABLE Produit
ADD CONSTRAINT fk_Produit_IDcat FOREIGN KEY (IDcat)
REFERENCES Categorie;`
 - Suppression de l'attribut Prenom dans CLIENT
`ALTER TABLE Client
DROP Prenom ;`

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

62

LDD – Drop

- Objectif
Suppression des données + de la structure d'une table
- Syntaxe
`DROP TABLE <table>;`
- Exemple
`DROP TABLE client ;`
- Remarque
Suppression uniquement des données (garde la structure)
`TRUNCATE TABLE client ;`
`DELETE FROM client ;`

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

63

Création de scripts

- JMerise <http://www.ifreesoft.com/JMerise/JMerise.zip>
 - Modélisation → scripts de création SQL

```
CREATE TABLE CANDIDAT(  
    N_Candidat      COUNTER NOT NULL ,  
    Nom             TEXT (25) ,  
    Prenom          TEXT (25) ,  
    DateNaissance   DATE ,  
    CONSTRAINT prk_constraint_CANDIDAT PRIMARY KEY (N_Candidat)  
);  
  
CREATE TABLE EXAMEN(  
    CodeExamen      TEXT (25) NOT NULL ,  
    Libelle         TEXT (25) ,  
    Type            TEXT (3) ,  
    CONSTRAINT prk_constraint_EXAMEN PRIMARY KEY (CodeExamen)  
);
```

2015-2016

M114 – Intro. BD – <http://denispallez.i3s.unice.fr>

64