

## Bases de Données 2

**Patricia Serrano Alvarado**

MC Université de Nantes

Laboratoire des Sciences du Numérique de  
Nantes (LS2N)

<https://pagespersowp.ls2n.fr/patriciaserrano/>

# Organisation

- ◉ 15h20 CM (11 créneaux)
- ◉ 13h20 TD (10 créneaux)
- ◉ 13h20 TP (10 créneaux)
  
- ◉ Contrôle de connaissances
  - ◉ 2 CC (coef 0,3)
  - ◉ 1 projet de TP en quadrinôme (coef 0,1)
  - ◉ 1 TP noté individuel (coef 0,1)
  - ◉ 1 examen 1h30 (coef 0,5)

# Equipe pédagogique

- Patricia Serrano Alvarado (MC). **CM, TD et TP**
- Matthieu Perrin (MC). **TD et TP**
- Richard Dufour (PU). **TP**

# Distribution des groupes

CM

Patricia Serrano Alvarado

TD

TD gpe 684 Patricia Serrano Alvarado

TD gpe 685 Patricia Serrano Alvarado

TD gpe 688;689 Matthieu Perrin

TP

TP 685 Patricia Serrano Alvarado

TP 684J Patricia Serrano Alvarado

TP 684I Matthieu Perrin

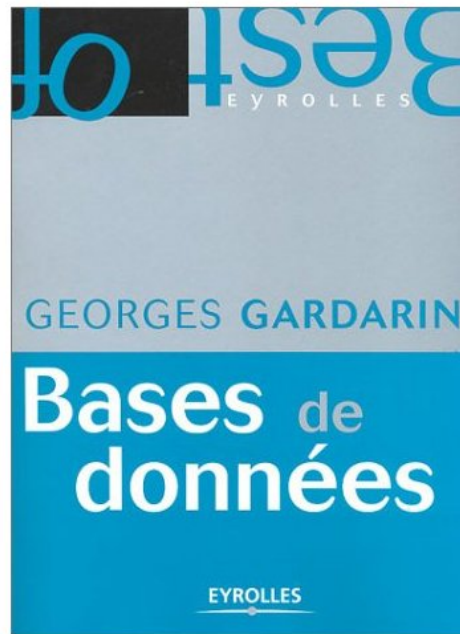
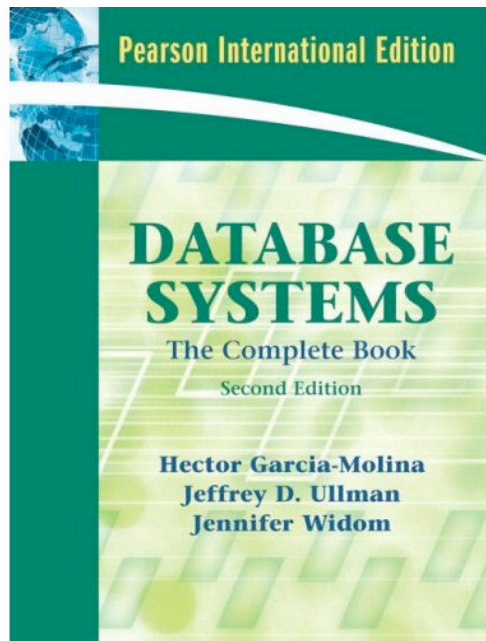
TP 688Q Matthieu Perrin

TP 689T Richard Dufour

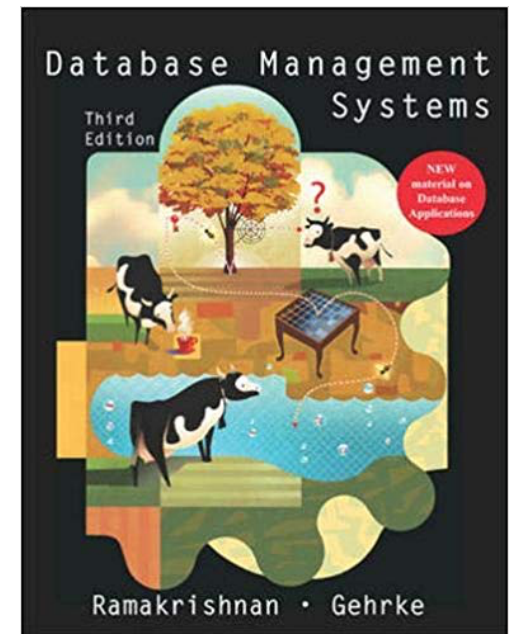
# Outils utilisés

- Madoc
  - Tout le matériel est disponible et organisé par semaine
  - Dépôts de rendus
  - Forum des nouvelles
- [MOOC Stanford](#)
- Oracle au CIE pour les TP (sous linux)
- [CIE à distance](#) : [documentation](#)
- Mail, selon le sujet, envoyer au responsable de CM, TD ou TP
  - Respecter les règles de bon sens
    - Pas écrire les weekends
    - Vous présenter
    - Relire votre mail et corriger les fautes d'orthographe

# Livres utilisés et disponibles à la BU



[Téléchargement gratuit](#)



# Déroulement

- 4 TP à faire et à rendre, **en binôme/trinôme**
- Premières quatre semaines
  - Concevoir un schéma de BD en bonne et due forme, **en quadrinôme (par groupe de TP)**
  - A rendre en semaine 5 (le vendredi)
  - **CC 1** en semaine 4
- Par la suite
  - Appliquer les concepts vus en CM/TD/TP à votre projet : fonctions PL/SQL, triggers, vues, indexes
  - **CC 2** en semaine 9
  - **Projet à rendre** + démo en semaine 11
  - **TP noté** individuel en semaine 12

# Votre projet de TD/TP

- Objectif : développer une base de données et y intégrer les aspects vus en cours et en TD
- Projet à faire en quadrinôme (par groupe de TP)
- 2 temps clés
  - 1 rapport sur votre schéma de BD (semaine 5)
  - 1 démonstration et rapport final (semaine 11)



# Programme du cours

1. Rappel cours BD1 (algèbre relationnelle, EA, modèle relationnel, SQL, etc.)
2. Formes normales et normalisation
3. Contrôle d'accès
4. PL/SQL
5. Triggers
6. Vues
7. Evaluation de requêtes
8. Stockage
9. Indexation
10. Transactions

# Aujourd'hui, un rappel de BD 1

- Qu'est-ce qu'une base de données ?
- Modèle Entité-Association
- Modèle Relationnel
- Algèbre relationnelle
- Langage SQL2

# Qu'est-ce qu'une BD ?

- Représentation du monde réel
- Collection de données structurées reliées par des relations
- Interrogeable et modifiable par des langages de haut niveau
  - Schéma
    - Organisation et types des données
  - Données
    - Le contenu/instances du schéma

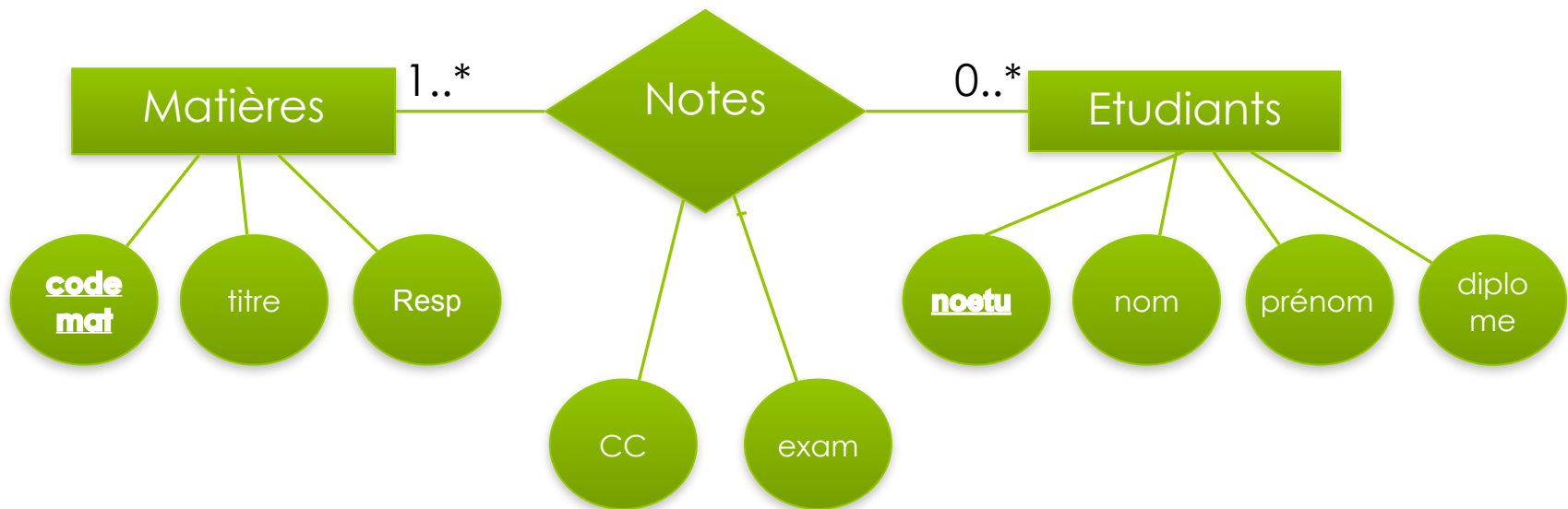
# Modèle Entité-Association (EA)

- Notions de
  - Entité
  - Attribut
  - Clé
  - Domaine
  - Association (cardinalité, degré, rôle)
  - Etc.

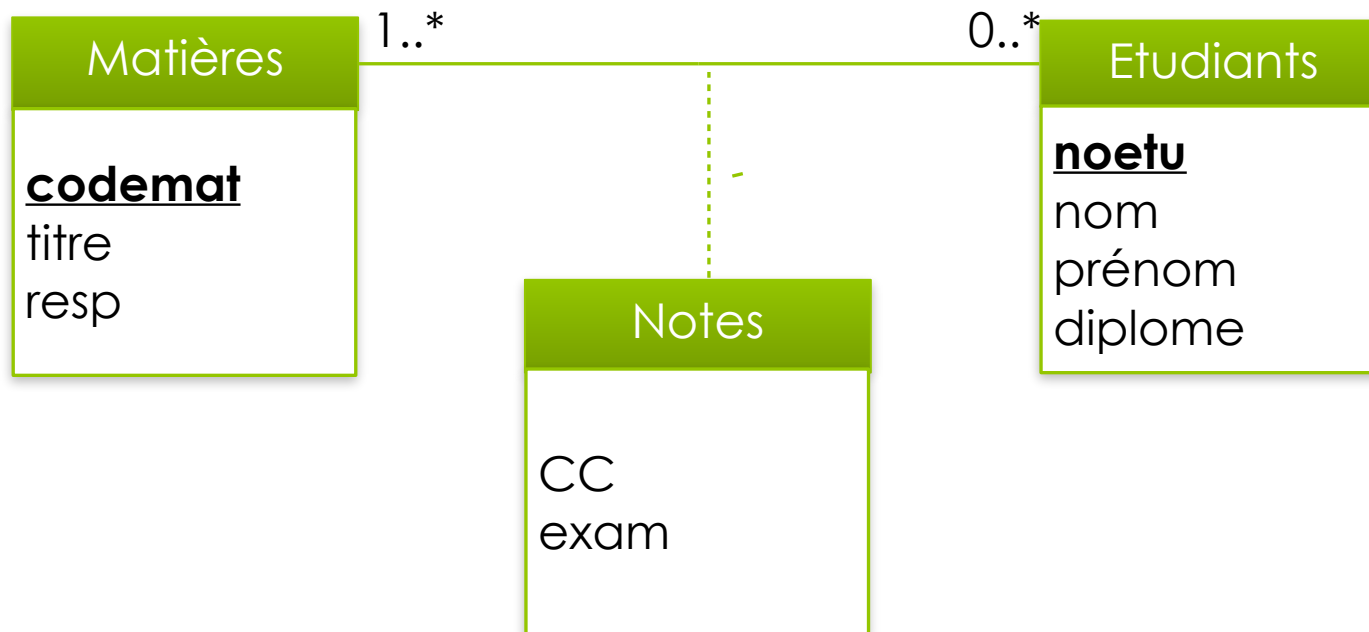
# Exemple de **modélisation** de BD

- Gérer les notes des étudiants par matière
- Entités
  - **Etudiants** (noetu, nom, prenom, diplome)
  - **Matières** (codemat, titre, resp)
- Association
  - **Notes** des étudiants (cc, exam) par matière

# Exemple de modèle EA



# Exemple de modèle EA en UML



# Modèle relationnel

- Notions de
  - Relation
  - Attribut
  - Clé
  - Domaine
  - Tuple
  - Schéma
  - Cardinalité d'une table :  $\text{card}(R)$  ou  $|R|$
  - Degré d'une table :  $\delta(R)$



# Transformation EA vers Relationnel (plusieurs règles)

- Entité -> relation
- Association -> relation avec clés étrangères
- Si cardinalité d'association 1:1 ou 1: 0..1 -> regrouper l'association avec l'entité d'origine en une seule relation

Etudiants

**noetu**: PK string  
nom: string  
prenom: string  
diplome: string

Notes

**noetu**: FK string  
**codemat**: FK string  
cc: real  
exam: real

Matieres

**codemat**: PK string  
titre: string  
resp: string

# Exemple d'instances

Quels degrés des tables ?  
Quelles cardinalités ?

Etudiants			
<u>noetu</u>	nom	prenom	diplôme
27845E	Dupont	Isabelle	L3 info
34561C	Legarec	Marc	L3 info
45678D	Martin	Robert	L3 info
64289C	Dupont	Letitia	L3 miage
23456E	Le Blanc	Michel	L3 miage
98076E	Robert	Christine	L3 miage

Notes			
<u>noetu</u>	<u>codemat</u>	cc	exam
27845E	X6I0050	15	12
27845E	X6I0010	10	11
27845E	X6I0030	11	13,5
34561C	X6I0050	10,2	11
34561C	X6I0010	10	13
34561C	X6I0030	12	10
45678D	X6I0050	10	8
45678D	X6I0030	5	5
64289C	X6IM020	15	10
64289C	X6IM010	12,5	10
64289C	X6I0050	13,1	12
23456E	X6IM010	7	2
23456E	X6I0080	14	13,8
98076E	X6IM020	12	12,8

Matières		
<u>codemat</u>	titre	resp
X6I0050	BD2	E512
X6I0010	Prog	E860
X6I0030	RO	E100
X6IM020	IHM	E350
X6IM010	Gestion	E670
X6I0080	Fichier	E900
X6I0040	Réseaux	E150

# Algèbre relationnelle

- Opérations de base unaires
  - Projection  $\pi$  (restriction sur les attributs)
  - Sélection  $\sigma$  (restriction sur les tuples selon un prédicat de sélection)
- Opérations de base ensemblistes
  - Union  $\cup$  (regroupement)
  - différence - (réduction)
  - produit cartésien  $\times$  (combinaison)
- Opérations dérivées
  - Intersection  $\cap$
  - complément  $\neg$
  - Jointures
    - naturelle  $\bowtie$
    - semi-jointure  $\ltimes, \rtimes$
    - interne  $\bowtie_Q$
    - externe full  $\bowtie_Q$ , left  $\ltimes_Q$ , right  $\rtimes_Q$  (ou symbole  $\bowtie_Q$ )
    - auto-jointure
- Opérations de mathématiques SUM, COUNT, AVG

Cours de Stanford sur l'algèbre relationnelle

<https://www.youtube.com/watch?v=tii7xcFiIOA>

# Equivalences algébriques

Commutativité et associativité de la jointure

$$\begin{aligned}E_1 \bowtie E_2 &= E_2 \bowtie E_1, \\(E_1 \bowtie E_2) \bowtie E_3 &= E_1 \bowtie (E_2 \bowtie E_3).\end{aligned}$$

Cascade de projections

$$\pi_{A_1, \dots, A_n}(\pi_{B_1, \dots, B_m}(E)) = \pi_{A_1, \dots, A_n}(E)$$

Cascade de sélections

$$\sigma_{F_1}(\sigma_{F_2}(E)) = \sigma_{F_1 \wedge F_2}(E)$$

Commutation sélection et projection

Si  $F$  ne porte que sur  $A_1, \dots, A_n$ ,

$$\pi_{A_1, \dots, A_n}(\sigma_F(E)) = \sigma_F(\pi_{A_1, \dots, A_n}(E))$$

Si  $F$  porte aussi sur  $B_1, \dots, B_m$ ,

$$\pi_{A_1, \dots, A_n}(\sigma_F(E)) = \pi_{A_1, \dots, A_n}(\sigma_F(\pi_{A_1, \dots, A_n, B_1, \dots, B_m}(E)))$$

# SQL : trois langages

- DDL (langage de définition de données)
  - Création de tables
  - Contraintes d'intégrité
    - De valeurs
    - D'unicité de clé
    - Référentielle (clé étrangère)
- DCL (langage de contrôle de données)
  - Contrôle d'accès
- DML (langage de manipulation de données)
  - Requêtes (*insert, select, update, delete*)

# Création d'une table (DDL)

```
CREATE TABLE notes(  
    noe varchar2(6) REFERENCES Etudiants (noetu),  
    codemat varchar2(6) REFERENCES Matieres,  
    noteex NUMBER(2,1),  
    notecc NUMBER(2,1),  
    CONSTRAINT notes_PK PRIMARY KEY(noe,codemat)  
);
```

# Cont.

- ALTER TABLE Notes ADD CONSTRAINT CHECK (noteex>=0 and noteex<21);
- INSERT INTO Notes VALUES ('99628C', 'MIAS215', 12, 15.5);
- INSERT INTO Notes (noe, codemat, noteex) VALUES ('28936E', 'MIAS216', 12);
- UPDATE Notes SET (notecc=10) WHERE noe = '28936E' AND codemat='MIAS216';

# Contrôle d'accès (DCL)

- `CREATE ROLE ROLE_ENSEIGNANT;`
- `CREATE ROLE ROLE_ETU;`
- `GRANT SELECT ON Notes TO ROLE_ETU;`
- `GRANT SELECT ON Matieres TO ROLE_ETU;`
- `GRANT SELECT, UPDATE ON Notes TO  
ROLE_ENSEIGNANT WITH GRANT OPTION;`



# Manipulation des données (DML)

```
SELECT *  
FROM notes NATURAL JOIN matieres;
```

```
SELECT *  
FROM etudiants e JOIN notes n ON  
e.noetu=n.noe ;
```

```
SELECT *  
FROM matières m LEFT OUTER JOIN notes n  
ON m.codemat=n.codemat;
```

Questions ?