

# TD6 : Vues, tables système et rappels en SQL

## V2.1.1

---



Cette œuvre de Mickaël Martin Nevot est mise à disposition sous licence Creative Commons Attribution - Utilisation non commerciale - Partage dans les mêmes conditions.

Document en ligne : [www.mickael-martin-nevot.com](http://www.mickael-martin-nevot.com)

---

## 1 Rappel : Prise en main d'un éditeur Oracle

Utilisez une des deux solutions ci-dessous.

### 1.1 Oracle Live SQL

Vous pouvez utiliser directement, sans installation ou configuration, l'interpréteur en ligne : <https://livesql.oracle.com>.

### 1.2 Oracle Cloud Free Tier

Mettez en place une solution d'hébergement en ligne d'un SGBD Oracle. Vous pouvez pour cela consulter le document Vade-Mecum mise en place d'un hébergement Oracle Cloud Free Tier.

Ajouter ensuite une base de données nommée `zenetude-bd`.

## 2 Rappel : Généralités

Voici la convention de nommage proposé dans cet enseignement :

- **mots clefs** : en lettre capitales (*upper case*) ;
- **relation** : première lettre de chaque mot en capitale (*pascal case*) ;
- **attributs** : premier mot en minuscule et première lettre de chaque mot suivant en capitale (*camel case*) ;
- **domaine** : en lettre capitales (*upper case*) au format `D_XXX`<sup>1</sup>.

Pour rappel, les valeurs saisies dans une base de données, comme les chaînes de caractères, sont sensibles à la casse et, par convention, sont saisies **en lettres capitales** (*upper case*), sans diacritique, dans le cadre de cet enseignement.

---

<sup>1</sup> Les domaines identiques sont surlignés avec la même couleur.

### 3 Rappel : Base de données exemple

La base de données exemple, Gestion pédagogique, utilisée dans les documents de travaux dirigés de cet enseignement, permet de gérer une formation en informatique, avec ses professeurs, ses étudiants, ses modules ainsi que leurs enseignements et notations.

#### 3.1 Dictionnaire de données

La base de données Gestion pédagogique a été élaborée à partir du dictionnaire des données suivant :

Tableau 1 – Dictionnaire des données de la base de données exemple

| Attribut    | Description                                           | Domaine <sup>2</sup>        | Remarques                      |
|-------------|-------------------------------------------------------|-----------------------------|--------------------------------|
| numEt       | Numéro d'un étudiant                                  | D_NUMET : NUMBER(6,0)       | Valeurs uniques                |
| nomEt       | Nom d'un étudiant                                     | D_NOM : VARCHAR2(30)        |                                |
| prenomEt    | Prénom d'un étudiant                                  | D_PRENOM : VARCHAR2(20)     |                                |
| cpEt        | Code postal d'un étudiant                             | D_CP : VARCHAR2(5)          |                                |
| villeEt     | Ville d'un étudiant                                   | D_VILLE : VARCHAR2(20)      |                                |
| anneeEt     | Année d'étude d'un étudiant                           | D_ANNEE : NUMBER(2,0)       | [1..2]                         |
| groupeEt    | Numéro de groupe d'un étudiant                        | D_GROUPE : NUMBER(1,0)      | [1..5]                         |
| numProf     | Numéro d'un professeur                                | D_NUMPROF : NUMBER(3,0)     | Valeurs uniques                |
| nomProf     | Nom d'un professeur                                   | D_NOM : VARCHAR2(20)        |                                |
| prenomProf  | Prénom d'un professeur                                | D_PRENOM : VARCHAR2(20)     |                                |
| adresseProf | Adresse d'un professeur                               | D_ADRESSE : VARCHAR2(40)    |                                |
| cpProf      | Code postal d'un professeur                           | D_CP : VARCHAR2(5)          |                                |
| villeProf   | Ville d'un professeur                                 | D_VILLE : VARCHAR2(20)      |                                |
| specProf    | Code d'une matière dont un professeur est spécialiste | D_CODE : VARCHAR2(7)        |                                |
| code        | Code d'un module                                      | D_CODE : VARCHAR2(7)        | Valeurs uniques                |
| Libelle     | Libellé d'un module                                   | D_LIBELLE : VARCHAR2(30)    |                                |
| heureCMPrev | Nombre d'heures de CM prévues pour un module          | D_NBHEURE : NUMBER(3,0)     |                                |
| heureCMReal | Nombre d'heures de CM réalisées pour un module        | D_NBHEURE : NUMBER(3,0)     |                                |
| heureTPPrev | Nombre d'heures de TP prévues pour un module          | D_NBHEURE : NUMBER(3,0)     |                                |
| heureTPReal | Nombre d'heures de TP réalisées pour un module        | D_NBHEURE : NUMBER(3,0)     |                                |
| discipline  | Discipline enseignée                                  | D_DISCIPLINE : VARCHAR2(15) | {INFORMATIQUE, MATHS, GESTION} |
| coefCC      | Coefficient du contrôle continu pour un module        | D_COEF : NUMBER(3,0)        | [0..100]                       |
| coefTest    | Coefficient du test pour un module                    | D_COEF : NUMBER(3,0)        | [0..100]                       |
| resp        | Numéro d'un professeur                                | D_NUMPROF : NUMBER(3,0)     |                                |

<sup>2</sup>Les types syntaxiques, utilisés pour la description des domaines, sont disponibles dans Oracle.

|          |                                                          |                      |         |
|----------|----------------------------------------------------------|----------------------|---------|
|          | responsable d'une matière                                |                      |         |
| codePere | Code du module père d'un module                          | D_CODE : VARCHAR2(7) |         |
| moyCC    | Moyenne de contrôle continu d'un étudiant dans un module | D_NOTE : NUMBER(2,0) | [0..20] |
| moyTest  | Moyenne de test d'un étudiant dans un module             | D_NOTE : NUMBER(2,0) | [0..20] |

### Remarques

Les modules forment une hiérarchie entre eux. Quel que soit leur niveau dans la hiérarchie, tous les modules possèdent les mêmes attributs. La racine de l'arbre ainsi formé par cette hiérarchie est le programme pédagogique national de formation en informatique. Il se décompose en véritables modules, eux-mêmes se décomposant en sous-modules et ainsi de suite jusqu'aux feuilles qui correspondent aux matières. Les matières sont les seuls modules pouvant être enseignés et susceptibles de recevoir des notes. Certains modules sont associés à une discipline.

Voici la hiérarchie des modules de l'extension de la base de données exemple représentée sous forme d'arborescence des codes :

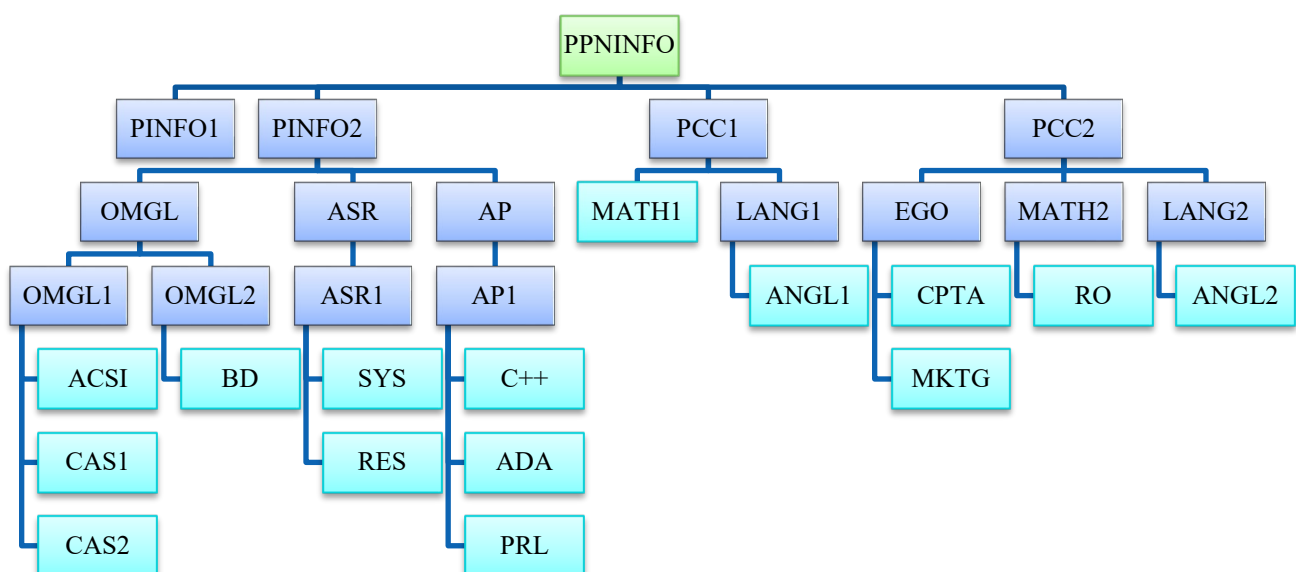


Figure 1 – Hiérarchie des modules

Les coefficients de contrôle continu et de test d'un module sont des pourcentages. Ils peuvent ne pas être renseignés. Si un des deux coefficients n'est pas renseigné, l'autre ne doit pas l'être non plus. Leur somme pour un même module est donc soit de 0, soit de 100.

Nous considérons qu'il n'y a pas deux personnes homonymes ayant le même prénom et le même nom.

## 3.1 MCD

Voici le MCD correspondant :

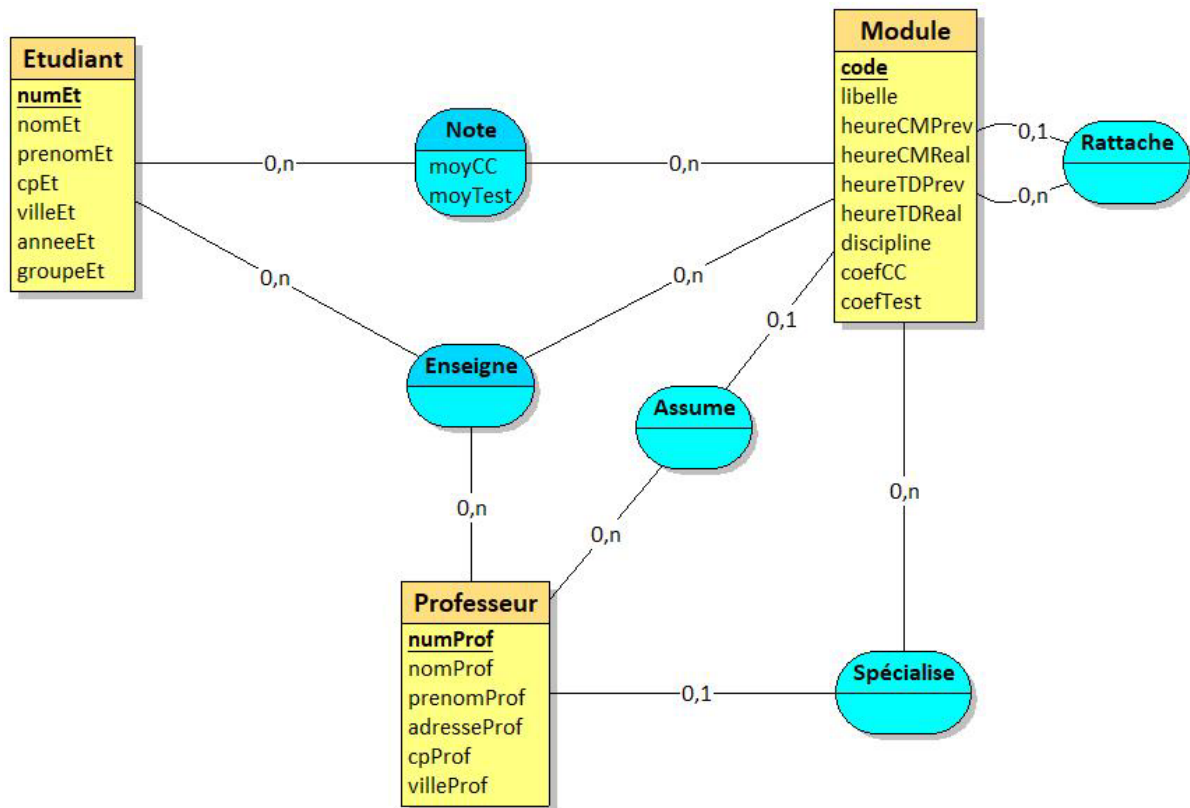


Figure 2 – Modèle conceptuel des données (MCD)

### 3.2 Schéma relationnel

Les clefs primaires sont soulignées et les clefs étrangères en *italique suivies d'un #*.

Le schéma relationnel de la base de données exemple est présenté ci-après :

```

Etudiant  (numEt, nomEt, prenomEt, cpEt, villeEt, anneeEt, groupeEt)
Professeur (numProf, nomProf, prenomProf, adresseProf, cpProf, villeProf,
            specProf #)
Module    (code, libelle, heureCMPprev, heureCMReal, heureTDPrev,
            heureTDReal, discipline, coefCC, coefTest, resp #, codePere #)
Note      (numEt #, code #, moyCC, moyTest)
Enseigne  (numEt #, code #, numProf #)
  
```

#### Remarques

La clef étrangère *resp* dans **Module** représente le numéro d'un professeur responsable d'un module<sup>3</sup> et fait référence à la clef primaire *numProf*.

La clef étrangère *specProf* dans **Professeur** représente le code d'une matière dont un professeur est spécialiste et fait référence à la clef primaire *code*.

La clef étrangère *codePere* dans la relation **Module** fait référence à la clef primaire *code*.

<sup>3</sup> Idéalement, il ne devrait être possible d'y avoir qu'un responsable d'une matière, pas de n'importe quel module. Par souci de simplification, cette contrainte n'a pas été prise en compte dans l'intension de la base de données exemple.

Les autres clefs étrangères font référence aux clefs primaires de même nom.

## 4 Requêtes avec SQL

Formulez, en SQL, sur la base de données exemple, les requêtes d'interrogation suivantes.

### 4.1 Création et manipulation de vues

Répondez aux questions suivantes sous forme de requêtes de création de vues, avant de consulter chaque vue par une requête d'interrogation ou des tentatives d'insertion, une valide et une invalide, pour les vues de mise à jour. Les vues à créer ont pour objectifs de créer de nouvelles relations logiques répondants à des besoins fréquents d'assurer la prise en compte des contraintes d'intégrité de la base de données ou de limiter la consultation des données pour certains utilisateurs.

Q1 : Quels sont les numéros, noms et prénoms des professeurs de deuxième année ? *3 attributs, 9 tuples*

**Créez des vues supprimant les calculs complexes des questions correspondantes du TD5 :**  
Interrogations avancées en SQL.

Q2 : À partir du nombre d'étudiants de chaque groupe de seconde année, donnez le nombre moyen d'étudiants. *TD5, Q5, 1 attribut, 1 tuple (9)*

Q3 : À partir du nombre d'étudiants à qui chaque professeur enseigne, quel est le plus grand nombre d'étudiants ? *TD5, Q6, 1 attribut, 1 tuple (45)*

Q4 : À partir de la somme d'heures de cours pour chaque discipline, donnez les codes et libellés des modules ainsi que le pourcentage de ces sommes d'heures par rapport au total des heures de cours dans la discipline correspondante. *TD5, Q7, 3 attributs, 9 tuples*

Q5 : À partir, du nombre de professeurs qui enseignent chaque module d'une part, et de la moyenne de test maximale de ces modules d'autre part, donnez les codes des modules, le nombre de professeurs qui les enseignent et la moyenne de test maximale de ces Modules. *TD5, Q8, 3 attributs, 6 tuples*

Q6 : À partir de la moins bonne et de la meilleure moyenne de test du module ACSI, donnez les numéros des étudiants et les écarts respectifs entre les éventuelles moyennes de test des étudiants et ces moyennes extrêmes. *TD5, Q22, 3 attributs, 29 tuples*

Q7 : Créer une vue de mise à jour permettant d'assurer que les discipline insérées ou modifiées ne peuvent être que Gestion, Informatique ou Maths.

Q8 : Créer une vue de mise à jour permettant d'assurer que les responsables d'un Module inséré ou modifié doivent enseigner ce Module.

## 4.2 Requêtes complexes3

Q9 : Quels sont les numéros, noms et prénoms des étudiants ayant eu une note dans le module Principes des BD ou dans un de ses sous-modules direct ou pas ? *3 attributs, 19 tuples*

Q10 : Donnez les libellés des modules et les libellés des modules d'où ils sont directement issus. *2 attributs, 30 tuples*

Q11 : Quels sont les groupes de seconde année, dans lesquels un étudiant a obtenu, pour le module Conception de SI, une meilleure note de test que le meilleur étudiant du groupe d'étudiants 3 dans le même Module ? *1 attribut, 1 tuple (4)*

## 4.3 Consultation des tables systèmes

En lieu et place du schéma relationnel de la base de données exemple, focalisons-nous plutôt à présent sur les vues système suivantes issues du dictionnaire de données Oracle dont un extrait est présenté ci-après :

```
User_Tables      (table_Name, tablespace_Name, ...)
User_Tab_Columns (table_Name, column_Name, data_Type, ..., data_Length, ...)
User_Cons_Columns (... , constraint_Name, table_Name, column_Name, ...)
User_Constraints (... , constraint_Name, constraint_Type, table_Name, ...)
User_Objects     (object_Name, ..., object_Type, ...)
```

### Remarques

Les valeurs de l'attribut `data_Type` correspondent aux types de données Oracle : `VARCHAR2`, `NUMBER`, `DATE`, etc.

La valeur de l'attribut `constraint_Type` peut être `P`, pour PRIMARY KEY, `R`, pour REFERENCES, `C` pour CHECK, etc.

La valeur de l'attribut `object_Type` peut être `TABLE`, `VIEW`, `FUNCTION`, `PROCEDURE`, `TRIGGER`, `INDEX`, etc.

Effectuez les requêtes sur les relations actuelles, pas celles éventuellement dans la corbeille. N'exécutez qu'une seule requête à la fois.

**En cas de doute ou de curiosité, n'oubliez pas d'utiliser la commande** `DESCRIBE`.

Q12 : Quels sont les noms des attributs de la relation `Professeur` ? *1 attribut, 7 tuples*

Q13 : Quelles sont les tables ? *1 attribut, 5 tuples*

- Q14 : Quelles sont les noms des contraintes d'intégrité ? *1 attribut, 20 tuples*
- Q15 : Quelles sont les noms des contraintes d'intégrité définies sur des relations ayant au moins un attribut de type `VARCHAR2` de 20 caractères ou plus ? *1 attribut, 13 tuples*
- Q16 : Quelles sont, pour les attributs de type `NUMBER`, les noms des contraintes et les noms des attributs sur lesquels elles portent ? *2 attributs, 15 tuples*
- Q17 : Quels sont les noms des contraintes, les noms attributs sur lesquels elles portent et le type de ces attributs, pour les relations ayant au moins un attribut de type `NUMBER` ? *3 attributs, 26 tuples*
- Q18 : Quelles sont les noms des contraintes de clefs primaires ? *1 attribut, 5 tuples*
- Q19 : Quels sont les noms des attributs, et leurs relations correspondantes, d'attributs portant le même nom dans des relations différentes ? *3 attributs, 7 tuples*