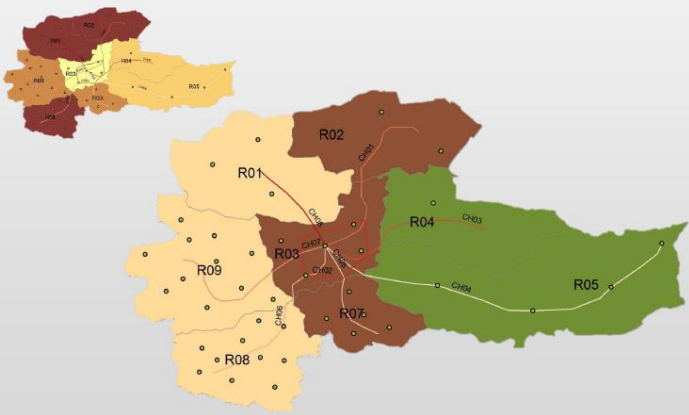




Gestion de bases de données

Généralités

Semestre 3/Unité : UET 311

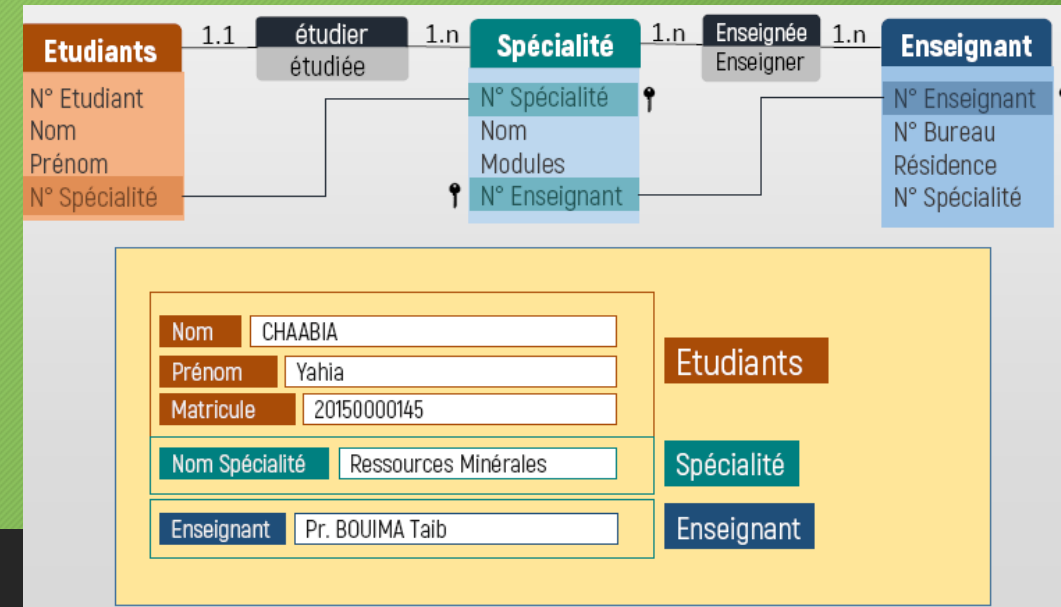
Coeff. 2 /Crédit. 2

Cours : 01H30 / TP : 01h30

C. SOULIMANE

Choukri.soulimane@univ-tlemcen.dz

Cours : 01H30 / TP : 01h30



Puits Pétrolier	Réservoirs	Chemin
N° Puits Code Puits Région	N° Réservoir Région N° Chemin	N° Chemin Code Distance à la base

FID	Shape *	Id	N° Puits
0	Point	0	0
1	Point	0	0
2	Point	0	0
3	Point	0	0
4	Point	0	0
5	Point	0	0
6	Point	0	0
7	Point	0	0
8	Point	0	0
9	Point	0	0
10	Point	0	0
11	Point	0	0
12	Point	0	0
13	Point	0	0
14	Point	0	0
15	Point	0	0
16	Point	0	0
17	Point	0	0
18	Point	0	0
19	Point	0	0

FID	Shape *	Id	N° Réservoir
0	Polygon	0	R01
1	Polygon	0	R02
2	Polygon	0	R03
3	Polygon	0	R04
4	Polygon	0	R05
5	Polygon	0	R07
6	Polygon	0	R06
7	Polygon	0	R09

FID	Shape *	Id	N° Chemin
0	Polyline	0	CH08
1	Polyline	0	CH01
2	Polyline	0	CH03
3	Polyline	0	CH07
4	Polyline	0	CH06
5	Polyline	0	CH05
6	Polyline	0	CH04
7	Polyline	0	CH02

- **Objectifs de l'enseignement**

- Ce cours permet aux étudiants de saisir, de traiter et de modéliser des données géographiques,
- spatialement localisées, par des logiciels.
- **Connaissances préalables recommandées**
- Savoir maîtriser l'outil informatique et les notions de base en géographie

1- Notion de bases de données

Les données permanentes constituent certainement le matériau de base à partir duquel vont s'élaborer la plupart des applications informatiques.

certaines de celles-ci se réduisent à gérer et consulter des données :

répertoire d'adresses, références bibliographiques ou catalogue de pièces de rechange. Ces données sont rangées dans ce qu'on appelle un fichier et sont structurées en enregistrements. Chaque correspondant du répertoire d'adresses y est décrit par un enregistrement qui reprend son nom et son prénom, ainsi que ses coordonnées postales et téléphoniques. De tels fichiers sont gérés par des logiciels simples et intuitifs et ne posent guère de problèmes à leurs utilisateurs.

1- Notion de bases de données

Les applications plus complexes réclament des données dont la structure est elle aussi plus complexe. Les données sont classées dans plusieurs fichiers en fonction des objets qu'elles décrivent : fichier des clients, fichier des produits, fichier des commandes, fichier des factures, etc. Il existe entre les fichiers des liens qui sont à l'image des relations entre les objets décrits. C'est via ces liens entre les données qu'on indique les commandes qui ont été émises par tel client, ou les produits qui sont référencés par telle commande. Il apparaît aussi que les données nécessaires à

1- Notion de bases de données

une application pourraient être utiles à d'autres applications, voire même à d'autres utilisateurs. Ces données constituent alors ce qu'on appelle une **base de données**.

Gérer de telles données n'est plus à la portée de logiciels élémentaires. Garantir la qualité des données enregistrées (retrouve-t-on ce qu'on a enregistré?), leur cohérence (le client de chaque commande est-il répertorié?), les protéger en cas d'incident, permettre à plusieurs utilisateurs d'y accéder simultanément, tout en contrôlant strictement l'accès aux données confidentielles, offrir de bonnes performances d'accès à toutes les applications, en particulier celles qui sont interactives, sont des fonctions qui réclament des logiciels puissants et complexes, les **Systèmes de gestion de bases de données, ou SGBD**

.

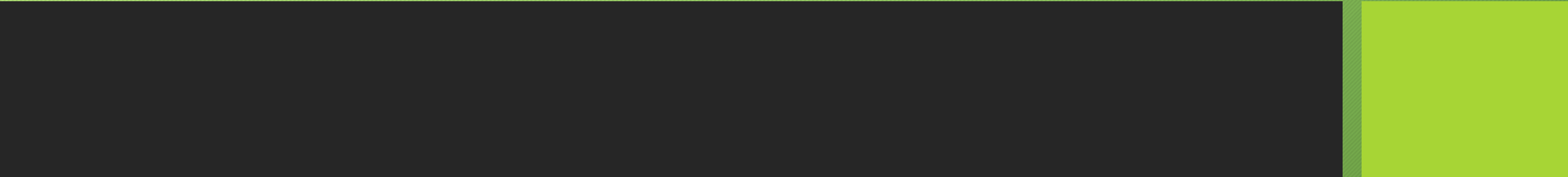
1- Notion de bases de données

Ce type de logiciels constitue l'un des outils fondamentaux de développement des grosses applications informatiques, mais aussi d'applications plus légères ou de serveurs Web. Ils existent sur toutes les plates-formes, depuis les gros ordinateurs (mainframes ou serveurs de grande puissance) jusqu'au PC portable ou au PDA, voire au téléphone mobile.

2- Les principes du relationnel

❑ *Principes*

Une **base de données relationnelle** apparaît comme une collection de tables de données, ou fichiers *plats*. Il s'agit d'une structure extrêmement simple et intuitive qui, pour l'utilisateur du moins, ne s'encombre d'aucun détail technique concernant les mécanismes de stockage sur disque et d'accès aux données. La figure 3.6 montre une petite base de données relationnelle constituée des tables CLIENT, COMMANDE, DETAIL et PRODUIT. Ce schéma, qui est suffisant pour décrire les données et leur structure, ne dit rien des techniques d'implantation de ces données sur disque par exemple.



Toutes les manipulations s'effectuent au moyen d'un unique langage, SQL (*Structured Query Language*). Ce langage permet à l'utilisateur de demander au SGBD de créer des tables, de leur ajouter des colonnes, d'y ranger des données et de les modifier, de consulter les données, de définir les autorisations d'accès. Les instructions de consultation des données sont essentiellement de nature déclarative et non procédurale.

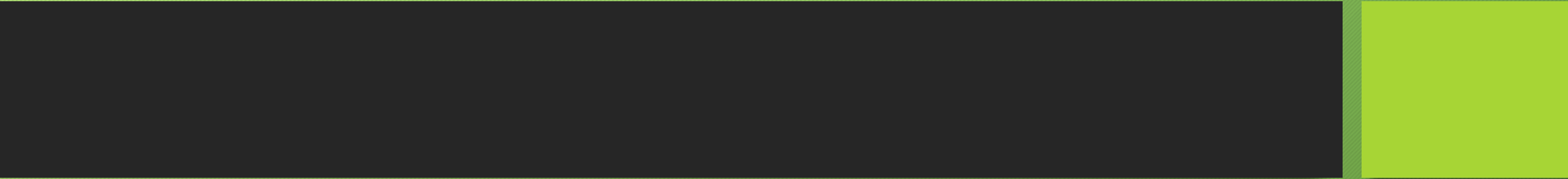
On y décrit les propriétés des données qu'on recherche, notamment en spécifiant une condition de sélection, mais on n'indique pas le moyen de les obtenir, décision qui est laissée à l'initiative du SGBD.

Le langage SQL semble donc destiné tant au développeur informaticien qu'à l'utilisateur final. Il est vrai que des requêtes simples s'expriment simplement.

Rechercher le nom et l'adresse de tous les clients de Tlemcen s'exprime par la requête suivante, qui, appliquée à la table CLIENT comportant les colonnes NOM, ADRESSE et LOCALITE, ne réclame guère d'effort d'interprétation :

```
select NOM, ADRESSE  
from CLIENT  
where LOCALITE = 'Tlemcen'
```

Bien sûr, l'extraction de données issues de plusieurs tables, et satisfaisant des conditions complexes, s'exprimera par des requêtes également plus complexes, qui nécessitent de la part de l'utilisateur qui les formule un apprentissage adéquat. On imagine aisément en effet que si on désire obtenir la liste des localités, accompagnées des quantités totales commandées par produit, et ordonnées par nombre décroissant de clients dont le compte est négatif, il nous faudra rédiger une requête beaucoup plus élaborée que celle qui vient d'être proposée.



Le marché offre aujourd'hui nombre d'outils qui permettent d'accéder à une base de données relationnelle sans qu'il soit nécessaire de maîtriser le langage SQL; citons notamment MS Access et FileMaker Pro. Ces modes d'accès sont cependant limités à des requêtes simples.