

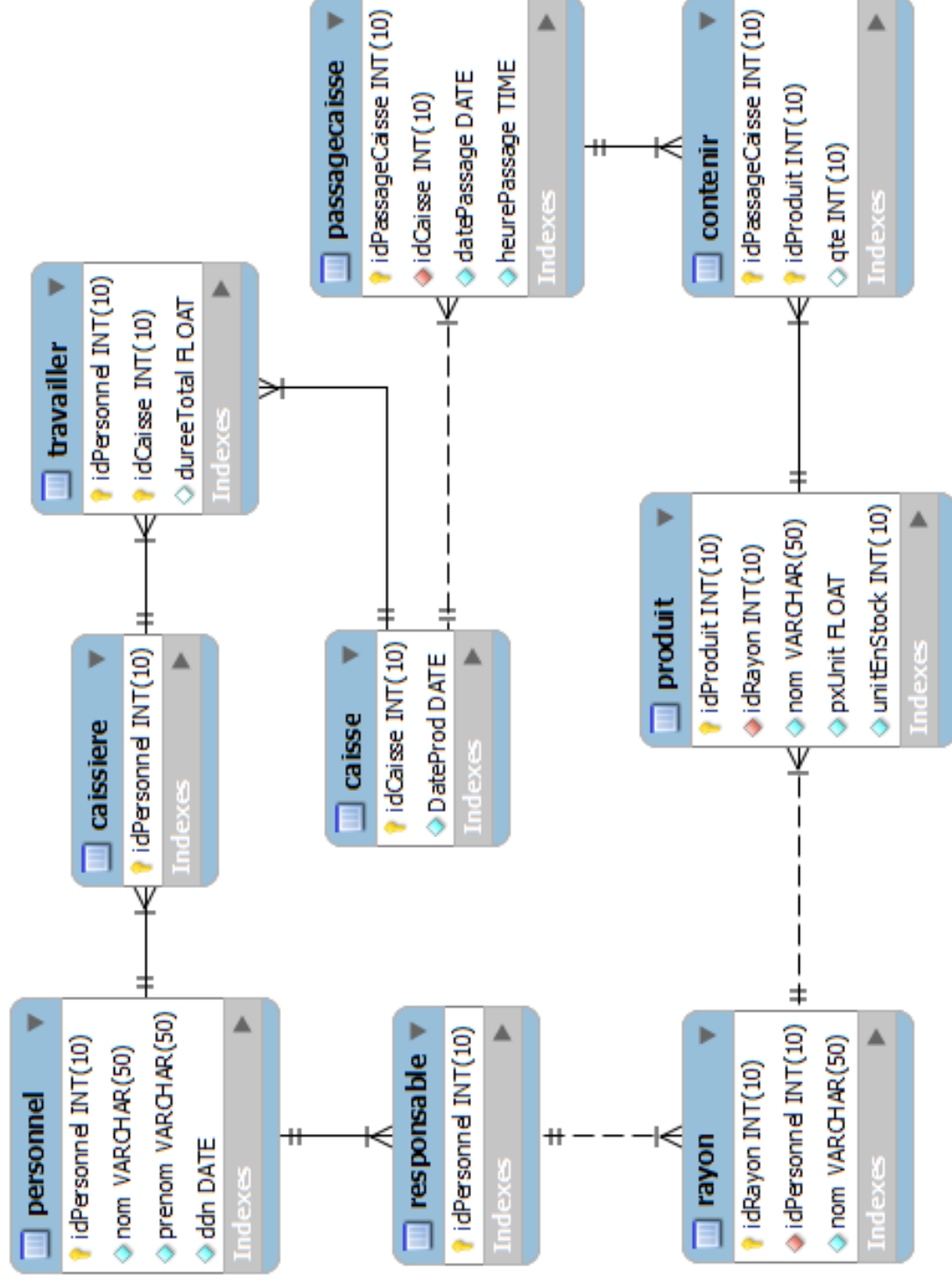
02 – Interroger une table

SI3 – Exploitation des données

Introduction

- SQL : Structured Query Language
- Syntaxe simple
- Interrogation avec SELECT:
 - Sélection de colonnes : projection
 - Sélection de certaines lignes : restriction
 - Combiner plusieurs tables : jointure
 - + union, intersection, différence relationnelle
- Possibilité de combiner toutes ces opérations
- Terminer instruction par ;

Exemple du cours



Sélection de colonnes

- Syntaxe

```
SELECT nomColonne1 , nomColonne2  
FROM uneTable ;
```

- Pour sélectionner toutes les colonnes : *

Sélection de colonnes – Exemple

- Nom et prix de tous les produits de la table
Produit :

```
SELECT nom , pxUnit  
FROM Produit ;
```

Cf. démonstration

nom	pxUnit
Livre	76
Chien	30
Chat	16
lapin	95
Eau	54
Tortue	8
Vin	99
CD Vierge	44
DVD Vierge	57
Agrafeuse	79
Lampe	3
Casque	57
PDA	95

La clause DISTINCT

- La clause distinct permet d'éliminer les doublons à l'affichage:

```
SELECT DISTINCT nom  
FROM Personnel ;
```

- N'affichera qu'une seule fois Imum alors que les 2 frères Max et Min sont dans la table.

La clause WHERE

- Permet de supprimer des lignes à l'affichage.
- Donne la possibilité de filtrer les données à retourner.
- Syntaxe :
 - `SELECT lesColonnes`
 - `FROM uneTable`
 - `WHERE lesPredicats;`

La clause WHERE : Exemple

- Nom des produits dont le prix est supérieur à 20€ :

```
SELECT nom
```

```
FROM Produit
```

```
WHERE pxUnit > 20 ;
```

- Le prédicat est évalué pour chaque ligne.
- La requête retourne uniquement les lignes pour lesquelles le prédicat est VRAI.

Opérateur dans un prédicat

- Prédicat :
 - `variableGauche op variableDroite`
- Opérateurs de comparaisons standards:
 - `=`
 - `!=`
 - `>`
 - `<`
 - `>=`
 - `<=`

Opérateurs particuliers : BETWEEN

- Expr BETWEEN expr1 AND expr2

SELECT nom

FROM Produit

WHERE pxUnit BETWEEN 16 AND 25 ;

- Retourne les produits (le nom) dont le prix est compris entre 16 et 25 €.

Opérateurs particuliers : IN

- Expr IN (unEnsembleDeValeurs)
SELECT pxUnit , UnitEnStock
FROM Produit
WHERE nom in ('Lampe' , 'Casque' , 'Sac');
- Affiche le prix et le nombre en stock des lampes, des casques, et des sacs.

Opérateurs particuliers : LIKE

- Ne s'applique que sur les chaînes de caractère
- Signifie « ressemble à »
- Utilise des caractères de substitution :
 - % : Remplace n'importe quel nombre de caractères, y compris aucun
 - _ : Remplace exactement un caractère

Opérateurs particuliers : LIKE

- Exemple :
 - `SELECT *`
 - `FROM Produit`
 - `WHERE nom like '%euse' ;`
- Donne la liste des produits (tous les champs) dont le nom finit par « euse ».

Expression gauche ou droite

- Dans les variables gauches ou droites :
 - Une valeur
 - Une opération
 - Le nom d'une colonne
 - Opérations sur des colonnes
 - Une sous-requête

Prédicats composés

- Possibilité de mettre plusieurs prédicats dans le WHERE.
- Combinés grâce aux opérateurs logiques:
 - AND : Et logique
 - OR : Ou logique
- AND prioritaire sur OR : utiliser des parenthèses
- Inverser valeur booléenne : NOT

Exemple plusieurs prédicats

```
SELECT nom , pxUnit  
FROM Produit  
WHERE idRayon IN ( 1 , 2 )  
      AND pxUnit < UnitEnStock ;
```

- Donne le nom et le prix des produits appartenant au rayon 1 ou bien au rayon 2. Ces produits doivent avoir un prix inférieur à la quantité en stock.

Exemple plusieurs prédicats

```
SELECT nom  
FROM Personnel  
WHERE nom like '_mum'
```

```
OR ddn < '1990/02/23' ;
```

- Donne le nom des employés qui sont nés avant le 23 février 1990 ou bien dont le nom de 4 lettres se termine par mum.

La valeur NULL

- NULL est affecté si aucune valeur n'a été donné à l'insertion de l'occurrence. (et pas de DEFAULT, ni AI)
- Possibilité d'affecter NULL à une valeur.
- Souvent la valeur par défaut
- Sélection avec is NULL
- Différent de la valeur 0 ou de la chaîne vide
- NULL + valeur n'a aucun sens

La valeur NULL : Exemple

```
SELECT nom , prenom  
FROM Personnel  
WHERE ddn is NULL ;
```

- Retourne les employés dont la date de naissance n'a pas été affectée

Renommer une colonne

- Utiliser des mots réservés
- Utilisé suite à une opération dans le SELECT
- On renomme avec AS ou rien
- Exemple :

```
SELECT pxUnit * 1.196 AS pxTTC  
FROM Produit ;
```

Classement des résultats

- Possibilité de classer les résultats à l'affichage
- Classement sur 1 ou plusieurs colonnes
- Classement ordre croissant ou ordre décroissant:
 - ASC : croissant (par défaut)
 - DESC : décroissant
- Clause ORDER BY toujours à la fin
- Ne classe pas les tables, uniquement la vue
- Valeurs NULL en tête (ASC ou DESC)

Classement des résultats : Exemple

```
SELECT nom , prenom  
FROM Personnel  
WHERE ddn > '1987/06/23'  
ORDER BY nom , prenom DESC ;
```

- Affiche les employés (leur nom et prénom) qui sont nés après le 23 juin 1987.
- Ces employés seront classés par ordre alphabétique sur le nom et inversement alphabétique sur le prénom lorsque les noms sont identiques.

ORDER BY sur les n° de colonnes

```
SELECT nom , prenom  
FROM Personnel  
WHERE ddn > '1987/06/23'  
ORDER BY 1 , 2 DESC ;
```

- Même résultat que précédemment

Exercices

- Nom et quantité en stock des produits.
- Liste des passages en caisse avant le 18 avril 2009
- Noms des produits dont les stocks sont compris entre 50 et 100, triés par ordre inverse de prix.
- Nom des employés dont le nom commence par « Ma » mais dont le prénom ne se termine pas par « on ».

Exercices

- Nom des employés nés en 1990 et 1991. Trier par ordre inversement alphabétique sur le nom et sur l'ordre croissant de naissance.
- Donner les noms des produits des rayons 3, 5 et 6, qui valent plus de la moitié de leur quantité en stock.

FIN

- Questions

