

# Bases de données – cours 5

## Introduction à la création et modification des tables

Catalin Dima

# Insertion d'un tuple

- ▶ Une requête d'insertion dans une table a la forme suivante :

```
INSERT INTO R(a1, a2...)  
VALUES (v1, v2...);
```

- ▶ Le schéma de la relation est à renseigner dans INSERT INTO.
- ▶ Les types d'attributs doivent correspondre aux valeurs à insérer.
- ▶ Exemple :

```
INSERT INTO Employés(Id, Nom, Prénom, Dept, Salaire)  
VALUES (123, 'Dima', 'Catalin', 'Info', 30);
```

# Insertion d'un ensemble de tuples

- ▶ On peut enchaîner une sélection avec une insertion, pour remplir une table avec les résultats d'une requête `SELECT` :

```
INSERT INTO R(a1, a2...)  
  SELECT ... FROM ... WHERE;
```

- ▶ Le schéma de la table résultat du `SELECT` doit être identique au schéma de la table dans laquelle les tuples seront insérés.
- ▶ Exemple :
  - ▶ Tables *ÉtudiantL3Info*(*Id*, *Nom*, *Prénom*, *MoyenneS1*, *MoyenneS2*) et *PassageEnM1*(*NoEtudiant*, *Nom*, *Prénom*).
  - ▶ Rajouter à la deuxième table les étudiants ayant le droit de s'inscrire en M1 :

```
INSERT INTO PassageEnM1(Id, Nom, Prénom)  
  SELECT Id AS NoEtudiant, Nom, Prénom  
  FROM ÉtudiantL3Info  
  WHERE (MoyenneS1+MoyenneS2) >= 20;
```

# Suppression de tuples

- ▶ Forme de la requête de suppression :

```
DELETE FROM R(a1, a2...)  
WHERE <condition>;
```

- ▶ Exemple :

- ▶ Mêmes tables que l'exemple précédent.
- ▶ Suppression de la table *PassageEnM1*(*NoEtudiant*, *Nom*, *Prénom*) des étudiants ayant été admis en M1 :

```
DELETE FROM PassageEnM1(Id, Nom, Prénom)  
WHERE (MoyenneS1+MoyenneS2) >= 20;
```

- ▶ Dans la clause WHERE on peut placer des sous-requêtes.

# Modifications de tuples

- ▶ Forme de la requête de modification :

```
UPDATE R
SET <affectation-multiple>
WHERE <condition>;
```

- ▶ On modifie les tuples, on ne modifie pas le schéma de la relation – à voir plus tard !
- ▶ L'affectation multiple s'écrit  
Nom-attribut = expression  
où l'expression doit avoir le même type que celui de l'attribut, et peut utiliser les noms d'autres attributs de la relation.
- ▶ La clause `WHERE` peut contenir des sous-requêtes.

# Exemple de modification de tuples

- ▶ Augmenter de 10kEuros les salaires des salariés du département 'd1' :

```
UPDATE Employés
SET Salaire = Salaire + 10
WHERE Dept = 'd1';
```

- ▶ Augmenter de 10kEuros les membres du projet 'jeu en réseau' :

```
UPDATE Employés
SET Salaire = Salaire + 10
WHERE Dept IN
  (SELECT Dept
   FROM Projets
   WHERE Nom = 'jeu en reseau');
```

# Création de tables

- ▶ Création de la table *Employés*(*Id, Nom, Prénom, Dept, Salaire*) :

```
CREATE TABLE Employes(  
  Id int(11),  
  Nom char(20),  
  Prenom char(20),  
  Dept char(5),  
  Salaire int(11) DEFAULT 20,  
  Anniversaire DATE  
);
```

- ▶ D'autres attributs peuvent apparaître dans chaque déclaration de nom de colonne, contrôlant :
  - ▶ l'unicité des valeurs,
  - ▶ la possibilité d'avoir des valeurs `NULL`
  - ▶ l'auto-incrémentation des valeurs en cours d'insertion dans la table,
  - ▶ la valeur par défaut,
  - ▶ la déclaration des clés, etc.

# Suppressions et modifications du schéma

- ▶ Supprimer une table d'une base de données :

```
DROP TABLE Projets;
```

- ▶ **Attention à l'emploi !**

- ▶ Rajouter ou supprimer un attribut ALTER TABLE, exemple :

```
ALTER TABLE Employes  
ADD Tel-bureau CHAR(10)  
DROP Anniversaire;
```