

# CREATION DE TABLES

```
CREATE TABLE [propriétaire.] nomTable
( nomColonne1 type [DEFAULT valeur] [contrainteColonne],
  nomColonne2 type [DEFAULT valeur] [contrainteColonne],
  ...
  nomColonneN type [DEFAULT valeur] [contrainteColonne],

[contrainte],
[contrainte], ...
);
```

Type => type de données

DEFAULT => valeur par défaut

# Type de données

- CHAR (N) : chaîne de caractères de longueur fixe = N
- VARCHAR2(N) : chaîne de caractères de longueur variable mais inférieure à N
- NUMBER (X), Number(X,Y)
  - X est le nombre de chiffres au total dont Y après la virgule
- DATE

# Différents types de contraintes

- **NOT NULL** : la valeur de la colonne est obligatoire
- **UNIQUE** : les valeurs d'une colonne ne sont pas redondantes
- **PRIMARY KEY** : Clé primaire
- **REFERENCES** et **FOREIGN KEY** : clé étrangère
- **CHECK** : domaine de définition d'une colonne (l'ensemble des valeurs possibles)

- Les contraintes peuvent être définies à deux niveaux
  - Niveau colonne
  - Niveau externe à la colonne

Syntaxe de la contrainte

**CONSTRAINT** nomContrainte type de la contrainte

**Exemple : une 1ere méthode de création de la table U**

**Create Table U**

( NU Number,

NOMU Varchar2(30) **Not NULL**,

Ville Varchar2(35),

**Constraint** pk\_U **Primary Key** (NU),

**Constraint** ck\_ville\_U **Check** (Ville in ('Lyon',  
'Marseille', 'Paris', 'Dijon'))

);

## Exemple : une 2ème méthode de création de U

- **Create Table U**

```
( NU Number Constraint pk_U Primary Key,  
  NOMU Varchar2(30) Not NULL,  
  Ville Varchar2(35),  
  Constraint ck_ville_U Check (Ville in ('Lyon',  
  'Marseille', 'Paris', 'Dijon'))  
);
```

## Exemple de la création de la table P

- **Create Table P**

(NP Number,

NOMP Varcher2(40) NOT NULL,

Couleur varchar2(25) Not Null,

Poids Number,

**Constraint pk\_P Primary Key** (NP),

**Constraint ck\_Couleur\_P check** (Couleur = 'Rouge'  
Or Couleur='Vert' or Couleur = 'Noir'),

**Constraint ck\_Poids\_P Check** (Poids between 15  
and 34)

);



# Exemple de la création de la table F

- **Create Table F**

( NF Number,  
NOMF varchar2(28),  
Statut varchar2(10),  
Ville varchar2(25),

**Constraint pk\_F Primary Key (NF)**  
);

# Exemple : 1ere méthode de création de la table PUF

- **Create Table PUF**

( NP Number,

NU Number, NF Number,

Quantite Number not null,

**Constraint pk\_PUF Primary key (NP, NU, NF),**

**Constraint fk\_NP\_PUF Foreign Key (NP) references P(NP),**

**Constraint fk\_NU\_PUF Foreign Key (NU) references U(NU),**

**Constraint fk\_NF\_PUF Foreign Key (NF) references F(NF)**

**);**

## Exemple : 2ème méthode de création de la table PUF

- **Create Table PUF**

( NP Number **constraint** pk\_PUF **REFERENCES** P(NP),

NU Number, NF Number,

Quantite Number not null,

**Constraint** pk\_PUF **Primary key** (NP, NU, NF),

~~**Constraint** fk\_NP\_PUF **Foreign Key** (NP) **references** P(NP),~~

**Constraint** fk\_NU\_PUF **Foreign Key** (NU) **references** U(NU),

**Constraint** fk\_NF\_PUF **Foreign Key** (NF) **references** F(NF)  
);

# Ajout de colonnes dans la structure d'une table existante

- **ALTER TABLE** uneTable **ADD**

(colonne1 type .....,  
Colonne2 type ... );

Exemple :

**ALTER TABLE** F **ADD** (TEL varchar2(30) );

# Modification des colonnes d'une table existante

- **ALTER TABLE** uneTable **MODIFY**

(colonne1 type ....,  
Colonne2 type ... );

Exemple :

**ALTER TABLE** F **MODIFY** (TEL varchar2(20) );

# Suppression des colonnes d'une table existante

- **ALTER TABLE** uneTable **DROP COLUMN**  
Colonne1, Colonne2, ...;

Exemple :

**ALTER TABLE F DROP TEL;**

**MISE A JOUR DES TABLES**

# Insertion de données

- Insertion d'1 seule ligne

```
Insert into nom_table [(col1, col2, ..., colN)] values  
(expression1, expression2, ...expressionN);
```

Il y a autant d'expressions que de colonnes.

La liste des colonnes à renseigner est facultative. Quand elle est omise, toutes les colonnes sont alors renseignées.

- Exemple

```
Insert into dept (deptno,dname) values (50, 'MANAGEMENT');
```

```
Insert into dept values (60, Logistics, BOSTON);
```

```
Commit;
```



# Insertion de données

- Insertion de plusieurs lignes à la fois

**Insert into** nom\_table [(col1, col2, ..., colN)]  
requete\_select

- Exemple

Soit GS une table vide (même structure que emp)  
vide dans la quelle on souhaite stocker les employés  
ayant un gros salaire (sal > 3000).

**Insert into** GS select \* from emp where SAL > 3000;  
Commit;

# Mise à jour des données

- **UPDATE** nom\_table **SET** col1 = expression1, col2 = expression2, ..., colj= expressionj  
**WHERE** condition;

Seules lignes satisfaisant la condition du where seront mises à jour.

- Exemple

**Update** emp **set** sal=sal\*1.2 **where** sal < 1500;

**Update** emp **set** sal=sal\*1.2 **where** job = 'CLERK';

**Update** dept **set** loc='NEW YORK' **where** where loc=BOSTON;

**Commit;**

# Suppression de lignes

- **DELETE from** nom\_table **where** condition;

Les lignes satisfaisant la condition sont supprimées dans nom\_table.

Exemple.

**Delete from** emp **where** job = 'ANALYST';

**Delete from** emp; on vide la table emp;

Commit;

# Suppression de tables

- **Drop table** nom\_table;

Exemple.

Drop table emp;

Commit;