11

Créer des vues

Objectifs

A la fin de ce chapitre, vous pourrez :

- **décrire une vue**
- **créer, modifier et supprimer une vue**
- **extraire des données depuis une vue**
- **insérer, mettre à jour et supprimer des données depuis une vue**
- **créer et utiliser une vue en ligne**
- **réaliser une analyse de type n-premiers**

Objets de base de données

Objet	Description
Table	Unité de stockage de base constituée de lignes et de colonnes
Vue	Représentation logique de sous-ensembles de données d'une ou de plusieurs tables
Séquence	Génère des valeurs de clé primaire
Index	Améliore les performances de certaines interrogations
Synonyme	Autre nom attribué à un objet

Définition d'une vue

Table EMPLOYEES :

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALA
100	Steven	King	SKING	515.123.4567	17-JUN-87	AD_FRES	240
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-89	AD_VP	170
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-93	AD_VP	170
103	Alexander	Hunold	AHUNO_D	590.423.4567	03-JAN-90	IT_PROG	90
104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-91	IT_PROG	60
107	Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-99	IT_PROG	42
124	Kevin	Mourgos	KMOURGOS	650.123.5234	16-NOV-99	ST_MAN	58
141	Trenna	Rajs	TRAJS	650.121.8009	17-OCT-95	ST_CLERK	35
142	Curtis	Davies	CDAVIES	650.121.2994	29-JAN-97	ST_CLERK	31
143	Randall	Matos	RMATOS	650.121.2874	15-MAR-98	ST_CLERK	26
144	Peter	Vargas	PVARGAS	650.121.2004	09-JUL-98	ST_CLERK	25
149	Eleni	Zlotkey	EZLOTKEY	011.44.1344.429018	29-JAN-00	SA_MAN	105
174	Ellen	Abel	EABEL	011.44.1644.429267	11-MAY-96	SA_REP	110
176	Jonathon	Taylor	JTAYLOR	011.44.1644.429265	24-MAR-98	SA_REP	86
178	Kimberely	Grant	KGRANT	011.44.1644.429263	24-MAY-99	SA_REP	70
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-SEP-87	AD_ASST	44
201	Michael	Hartstein	MHARTSTE	515.123.5555	17-FEB-96	MK_MAN	130
202	Pat	Fay	PFAY	603.123.6666	17-AUG-97	MK_REP	60
205	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-94	AC_MGR	120
206	William	Gietz	WGIETZ	515.123.8181	07-JUN-94	AC_ACCOUNT	83

20 rows selected.

EMPLOYEE_ID	LAST_NAME	SALARY
149	Zlotkey	10500
174	Abel	11000
176	Taylor	8600

Avantages des vues

- Limitent l'accès aux données
- Facilitent la création d'interrogations complexes
- Garantissent l'indépendance des données
- Présentent les mêmes données sous différentes vues

Vues simples et vues complexes

Caractéristiques	Vues simples	Vues complexes
Nombre de tables	Une	Une ou plusieurs
Fonctions	Non	Oui
Groupes de données	Non	Oui
Opérations LMD sur une vue	Oui	Pas toujours

Créer une vue

- Imbriquez une sous-interrogation dans l'instruction `CREATE VIEW`.

```
CREATE [OR REPLACE] [FORCE|NOFORCE] VIEW view
  [(alias[, alias]...)]
 AS subquery
[WITH CHECK OPTION [CONSTRAINT constraint]]
[WITH READ ONLY [CONSTRAINT constraint]];
```

- La sous-interrogation peut contenir une syntaxe `SELECT` complexe.

Créer une vue

- Créez la vue **EMPVU80** qui doit contenir des informations sur les employés du service 80.

```
CREATE VIEW  empvu80
 AS SELECT  employee_id, last_name, salary
    FROM    employees
    WHERE   department_id = 80;
View created.
```

- Décrivez la structure de la vue à l'aide de la commande *i*SQL*Plus **DESCRIBE**.

```
DESCRIBE empvu80
```

Créer une vue

- Créez une vue en utilisant des alias de colonne dans la sous-interrogation.

```
CREATE VIEW  salvu50
 AS SELECT  employee_id ID_NUMBER, last_name NAME,
            salary*12 ANN_SALARY
    FROM    employees
    WHERE   department_id = 50;
View created.
```

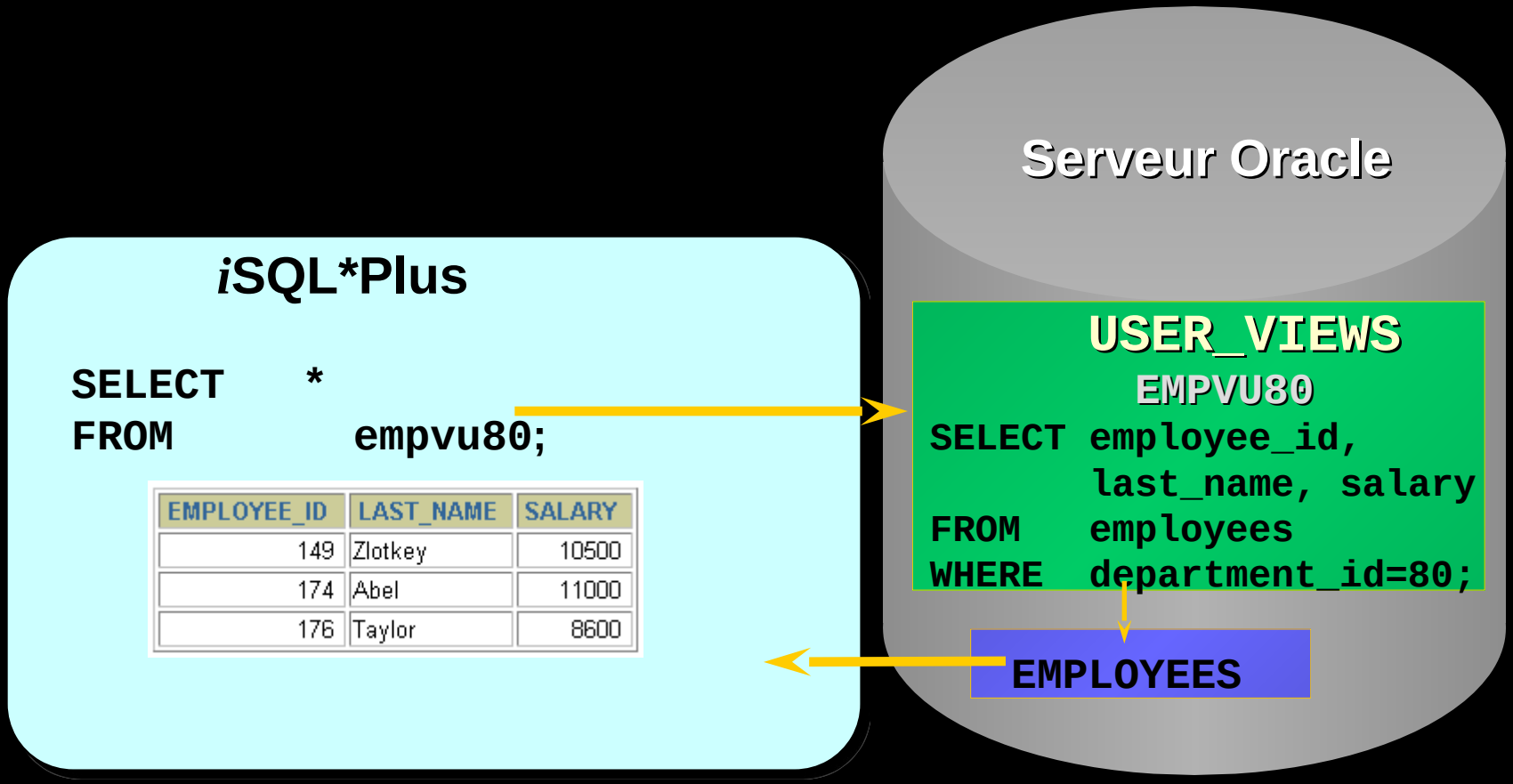
- Sélectionnez les colonnes de cette vue par leur nom d'alias.

Extraire des données d'une vue

```
SELECT *
FROM  salvu50;
```

ID_NUMBER	NAME	ANN_SALARY
124	Mourgos	69600
141	Rajs	42000
142	Davies	37200
143	Matos	31200
144	Vargas	30000

Interroger une vue



Modifier une vue

- Modifiez la vue `EMPVU80` à l'aide de la clause `CREATE OR REPLACE VIEW`. Ajoutez un alias pour chaque nom de colonne.

```
CREATE OR REPLACE VIEW empvu80
  (id_number, name, sal, department_id)
AS SELECT  employee_id, first_name || ' ' || last_name,
           salary, department_id
   FROM    employees
   WHERE   department_id = 80;
View created.
```

- Les alias de colonne de la clause `CREATE VIEW` s'affichent dans le même ordre que les colonnes de la sous-interrogation.

Créer une vue complexe

Créez une vue complexe contenant des fonctions de groupe pour afficher des valeurs provenant de deux tables.

```
CREATE VIEW dept_sum_vu
  (name, minsal, maxsal, avgsal)
AS SELECT    d.department_name, MIN(e.salary),
             MAX(e.salary),AVG(e.salary)
   FROM      employees e, departments d
   WHERE     e.department_id = d.department_id
   GROUP BY  d.department_name;
View created.
```

Règles d'exécution des opérations LMD sur une vue

- Vous pouvez exécuter des opérations LMD sur des vues simples.
- Vous ne pouvez pas supprimer une ligne si la vue contient :
  - des fonctions de groupe,
  - une clause `GROUP BY`,
  - le mot-clé `DISTINCT`,
  - la pseudo-colonne `ROWNUM`.

Règles d'exécution des opérations LMD sur une vue

Vous ne pouvez pas modifier les données d'une vue si elle contient :

- des fonctions de groupe,
- une clause `GROUP BY`,
- le mot-clé `DISTINCT`,
- la pseudo-colonne `ROWNUM`,
- des colonnes définies par des expressions.

Règles d'exécution des opérations LMD sur une vue

Vous ne pouvez pas ajouter de données dans une vue si celle-ci comporte :

- des fonctions de groupe,
- une clause `GROUP BY`,
- le mot-clé `DISTINCT`,
- la pseudo-colonne `ROWNUM`,
- des colonnes définies par des expressions,
- des colonnes `NOT NULL` se trouvant dans les tables de base qui ne sont pas sélectionnées par la vue.

Utiliser la clause WITH CHECK OPTION

- Vous pouvez vous assurer que les opérations LMD effectuées sur la vue restent dans le domaine de la vue à l'aide de la clause WITH CHECK OPTION.

```
CREATE OR REPLACE VIEW empvu20
AS SELECT   *
   FROM     employees
   WHERE    department_id = 20
   WITH CHECK OPTION CONSTRAINT empvu20_ck ;
View created.
```

- Toute tentative de modification du numéro de service dans une ligne de la vue échouera, car elle transgresse la contrainte WITH CHECK OPTION.

Refuser des opérations LMD

- Aucune opération LMD ne pourra être exécutée si vous ajoutez l'option `WITH READ ONLY` dans la définition de votre vue.
- Le serveur Oracle envoie un message d'erreur lors de toute tentative d'exécution d'une instruction LMD sur une ligne de la vue.

Refuser des opérations LMD

```
CREATE OR REPLACE VIEW empvu10
    (employee_number, employee_name, job_title)
AS SELECT   employee_id, last_name, job_id
   FROM     employees
   WHERE    department_id = 10
   WITH READ ONLY;
View created.
```

Supprimer une vue

La suppression d'une vue n'entraîne pas la perte des données, car toute vue est basée sur des tables sous-jacentes de la base de données.

```
DROP VIEW view;
```

```
DROP VIEW empvu80;
View dropped.
```

Vues en ligne

- Une vue en ligne est une sous-interrogation intégrant un alias (ou nom de corrélation) que vous pouvez utiliser dans une instruction SQL.
- Une sous-interrogation nommée, contenue dans la clause `FROM` de l'interrogation principale, est un exemple de vue en ligne.
- Une ligne en vue n'est pas un objet de schéma.

Analyse de type n-premiers

- Les interrogations n-premiers vous permettent d'identifier les *n* valeurs les plus petites ou les plus grandes présentes d'une colonne.
  Par exemple :
  - Quels sont les 10 produits les mieux vendus ?
  - Quels sont les 10 produits les moins vendus ?
- Les ensembles de valeurs les plus grandes et les plus petites correspondent à des interrogations n-premiers.

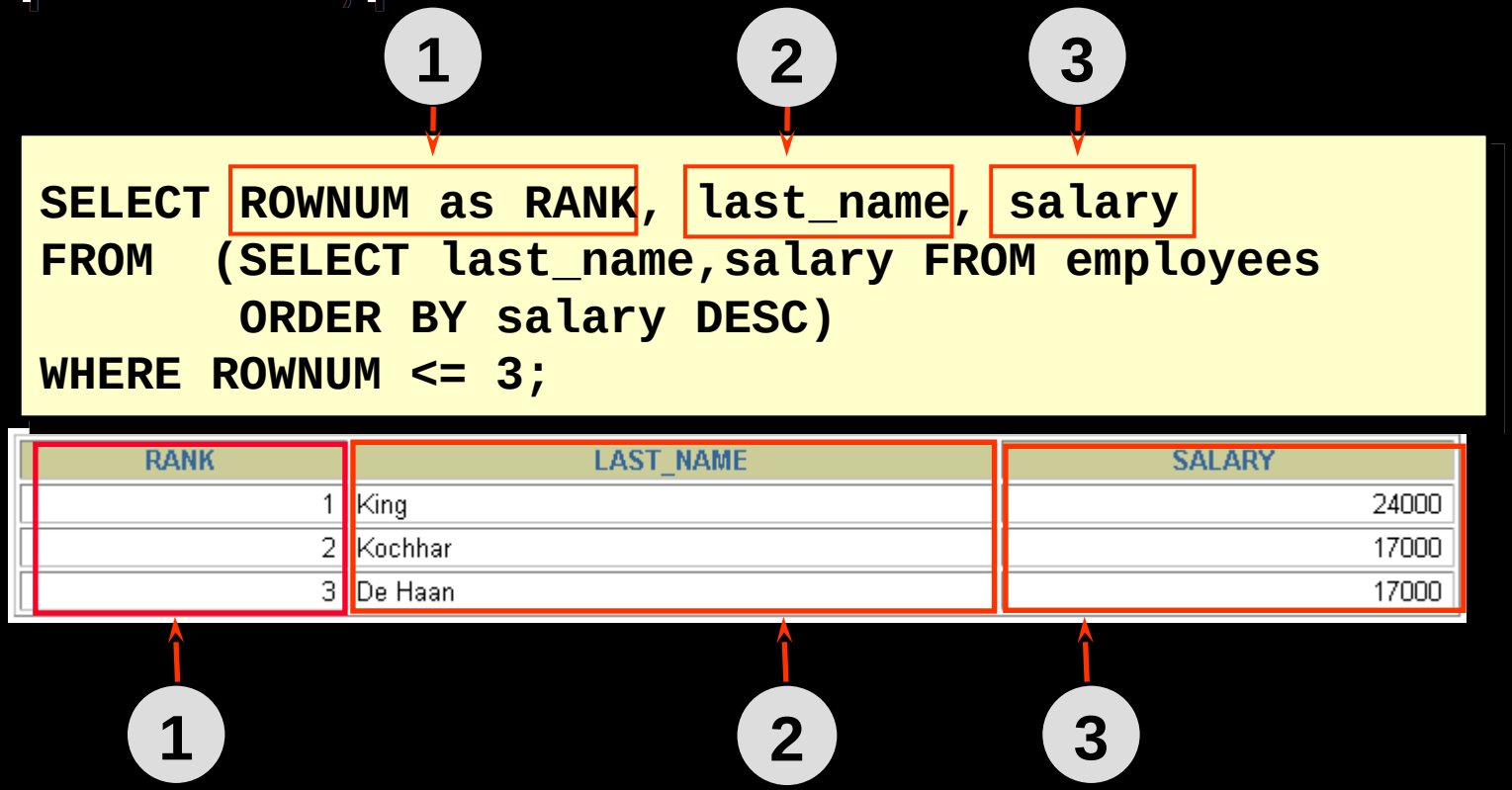
Réaliser une analyse de type n-premiers

La structure de haut niveau d'une analyse de type n-premiers se présente comme suit :

```
SELECT [column_list], ROWNUM
FROM   (SELECT [column_list]
        FROM table
        ORDER  BY Top-N_column)
WHERE  ROWNUM <=  N;
```

Exemple d'analyse de type n-premiers

Pour afficher le nom et le salaire des trois employés de la table EMPLOYEES qui touchent les salaires les plus élevés, procédez comme suit :



Synthèse

Ce chapitre vous à permis d'apprendre qu'une vue est issue de données provenant d'autres tables ou vues. Elle présente les avantages suivants :

- Elle restreint l'accès à la base de données.
- Elle simplifie les interrogations.
- Elle garantit l'indépendance des données.
- Elle permet de visualiser les mêmes données sous différentes formes.
- Elle peut être supprimée sans perte des données sous-jacentes.
- Une vue en ligne est une sous-interrogation comportant un alias.
- Une analyse de type n-premiers peut être réalisée à l'aide de sous-interrogations et d'interrogations externes.

Présentation de l'exercice 11

Dans cet exercice, vous allez :

- **créer une vue simple**
- **créer une vue complexe**
- **créer une vue avec une contrainte CHECK**
- **tenter de modifier des données dans une vue**
- **afficher des définitions de vue**
- **supprimer des vues**