Mettre des résultats en forme avec iSQL*Plus

Objectifs

A la fin de ce chapitre, vous pourrez :

- créer des interrogations nécessitant une variable de substitution
- personnaliser l'environnement iSQL*Plus
- afficher des résultats plus lisibles
- créer et exécuter des fichiers script

Variables de substitution

```
Je souhaite
                                      interroger différentes
... salary = ? ...
                                      valeurs.
  department_id = ? ...
... last_name = ? ...
                         Utilisateur
```

Variables de substitution

Utilisez des variables de substitution iSQL*Plus pour :

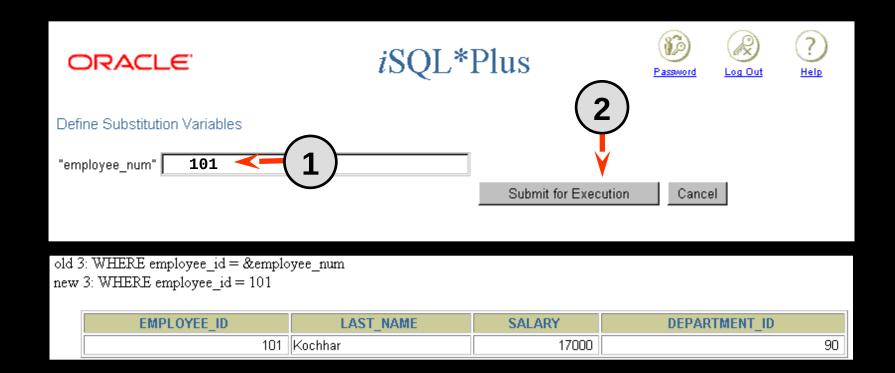
- stocker temporairement des valeurs,
 - Esperluette simple (&)
 - Esperluette double (&&)
 - Commande DEFINE
- transmettre des variables entre des instructions SQL,
- modifier dynamiquement des en-têtes et pieds de page.

Utiliser la variable de substitution &

Pour inviter l'utilisateur à entrer une valeur, utilisez une variable précédée d'une esperluette simple (&).

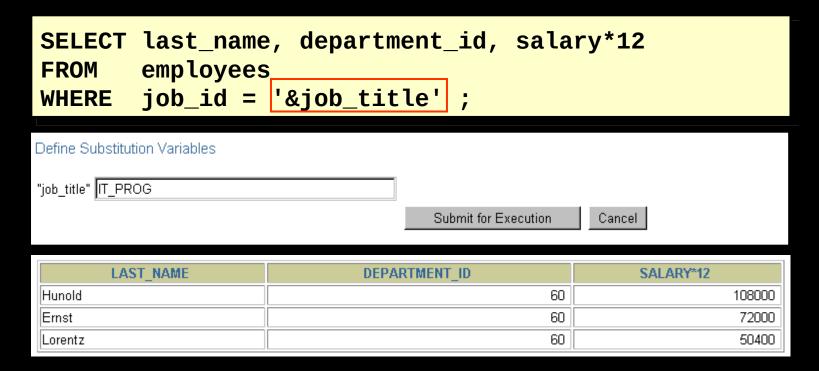
SELECT FROM WHERE	<pre>employee_id, employees employee_id =</pre>			depar	tment	t_id
ORACLE	≣'	<i>i</i> SQL*Plus		Password	Log Out	? Help
Define Substitution "employee_num"	Variables	Submit	t for Execution	Cance	ī	

Utiliser la variable de substitution &



Valeurs de type caractère et date dans les variables de substitution

Placez les valeurs de type caractère et date entre apostrophes.



Remplacer des noms de colonne, des expressions et du texte

Les variables de substitution peuvent être utilisées dans :

- les conditions WHERE,
- les clauses ORDER BY,
- les expressions de colonne,
- les noms de table,
- les instructions SELECT.

Remplacer des noms de colonne, des expressions et du texte

FROM en &c WHERE &c	ployee_id, las olumn_name ployees ondition order_column;	t_name, job_	_id,
Define Substitution Variables			
"column_name" salary			
"condition" salary > 15000			
"order_column" last_name			
		Submit for Execution	Cancel
EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
102	De Haan	AD_VP	17000
100	King	AD_PRES	24000
101	Kochhar	AD VP	17000

Définir des variables de substitution

 Vous pouvez prédéfinir des variables à l'aide de la commande DEFINE d'iSQL*Plus.

DEFINE *variable* = *value* crée une variable utilisateur dont le type de données est CHAR.

- Si vous devez prédéfinir une variable contenant des espaces, placez-la entre apostrophes lorsque vous utilisez la commande DEFINE.
- Vous pouvez utiliser une variable définie pour la session.

Commandes DEFINE et UNDEFINE

- Une variable reste définie jusqu'à ce que vous :
 - utilisiez la commande UNDEFINE pour l'effacer,
 - quittiez iSQL*Plus.
- La commande DEFINE vous permet de vérifier vos modifications.

```
DEFINE job_title = IT_PROG
DEFINE job_title
DEFINE JOB_TITLE = "IT_PROG" (CHAR)
```

```
UNDEFINE job_title
DEFINE job_title
SP2-0135: symbol job_title is UNDEFINED
```

Utiliser la commande DEFINE avec la variable de substitution &

 Créez une variable de substitution à l'aide de la commande DEFINE.

```
DEFINE employee_num = 200
```

 Faites précéder la variable d'une esperluette (&) pour remplacer la valeur dans l'instruction SQL.

```
SELECT employee_id, last_name, salary, department_id
FROM employees
WHERE employee_id = &employee_num;
```

EMPLOYEE_ID	LAST_NAME	SALARY	DEPARTMENT_ID
200 Whalen		4400	10

Utiliser la variable de substitution &&

Entrez une esperluette double (&&) si vous souhaitez réutiliser la valeur de la variable sans interroger l'utilisateur à chaque fois.

FROM <u>e</u>	mpl	oyee_id, oyees umn_name;	last_nam	e, job_io	d, &&colu	mn_name
Define Substitution Va						
				Submit for Execu	Cancel	
EMPLOYEE_ID		LAST_NAME	J(DB_ID	DEPARTMEN	IT_ID
	200	Whalen	AD_ASST	_		10
	201	Hartstein	MK_MAN			20
20						



Utiliser la commande VERIFY

Utilisez la commande VERIFY pour faire basculer l'affichage de la variable de substitution avant et après le remplacement des variables par des valeurs.

```
SET VERIFY ON

SELECT employee_id, last_name, salary, department_id

FROM employees

WHERE employee_id = &employee_num;

"employee_num" 200

Old 3: WHERE employee_id = &employee_num

new 3: WHERE employee_id = 200
```

Personnaliser l'environnement iSQL*Plus

 Utilisez les commandes SET pour contrôler la session en cours.

SET system_variable value

 Vérifiez au moyen de la commande SHOW les paramètres définis.

SET ECHO ON

SHOW ECHO

echo ON



Variables de la commande SET

ARRAYSIZE {20 | n}
 FEEDBACK {6 | n | OFF | ON}
 HEADING {OFF | ON}
 LONG {80 | n} | ON | text}

```
SET HEADING OFF
```

SHOW HEADING HEADING OFF

Commandes de mise en forme d'iSQL*Plus

- COLUMN [column option]
- TTITLE [text | OFF | ON]
- BTITLE [text | OFF | ON]
- BREAK [ON report_element]

Commande COLUMN

Cette commande contrôle l'affichage d'une colonne :

COL[UMN] [{column|alias} [option]]

- CLE[AR] Efface la mise en forme de la colonne
- HEA[DING] text Définit l'en-tête de colonne
- FOR[MAT] format Modifie le format d'affichage d'une colonne en fonction d'un modèle
- NOPRINT | PRINT
- NULL



Utiliser la commande COLUMN

Créez des en-têtes de colonne.

```
COLUMN last_name HEADING 'Employee|Name'
COLUMN salary JUSTIFY LEFT FORMAT $99,990.00
COLUMN manager FORMAT 999999999 NULL 'No manager'
```

 Affichez le paramétrage en cours de la colonne LAST_NAME.

```
COLUMN last_name
```

Annulez le paramétrage de la colonne LAST_NAME.

COLUMN last name CLEAR



Modèles de format pour la commande COLUMN

Elément	Description	Exemple	Résultat
9	Suppression du zéro de gauche	999999	1234
0	Affiche les zéros de tête	099999	001234
\$	Signe dollar flottant	\$9999	\$1234
L	Symbole monétaire local	L9999	L1234
	Position du signe décimal	9999.99	1234.00
,	Séparateur des milliers	9,999	1,234

Utiliser la commande BREAK

Utilisez la commande BREAK pour supprimer les doublons.

BREAK ON job_id



Utiliser les commandes TTITLE et BTITLE

Affichez des en-têtes et pieds de pages.

TTI[TLE] [text|OFF|ON]

Définissez un en-tête d'état.

TTITLE 'Salary|Report'

Définissez un pied de page d'état.

BTITLE 'Confidential'

Créer un fichier script pour exécuter un état

- 1. Créez et testez l'instruction SQL SELECT.
- 2. Enregistrez l'instruction SELECT dans un fichier script.
- 3. Chargez le fichier script dans un éditeur.
- 4. Ajoutez des commandes de mise en forme avant l'instruction SELECT.
- 5. Vérifiez que l'instruction SELECT est suivie du caractère de fin.

Créer un fichier script pour exécuter un état

- 6. Effacez les commandes de mise en forme après l'instruction SELECT.
- 7. Enregistrez le fichier script.
- 8. Chargez le fichier script dans la fenêtre de texte d'iSQL*Plus, puis cliquez sur le bouton Execute.

Exemple d'état

Fri Sep 28		⊏mpioyee Report	page 1
	1-1-		

Job Category	Employee	Salary
AC_ACCOUNT	Gietz	\$8,300.00
AC_MGR	Higgins	\$12,000.00
AD_ASST	Whalen	\$4,400.00
IT_PROG	Ernst	\$6,000.00
	Hunold	\$9,000.00
	Lorentz	\$4,200.00
MK_MAN	Hartstein	\$13,000.00
MK_REP	Fay	\$6,000.00
SA_MAN	Zlotkey	\$10,500.00
SA_REP	Abel	\$11,000.00
	Grant	\$7,000.00
	Taylor	\$8,600.00

Confidential





Sample Report

Fri Sep 28	⊨mployee	page 1
1 11 Sep 20	Report	page i

Job Category	Employee	Salary
AC_ACCOUNT	Gietz	\$8,300.00
AC_MGR	Higgins	\$12,000.00
AD_ASST	Whalen	\$4,400.00
IT_PROG	Ernst	\$6,000.00
	Hunold	\$9,000.00
	Lorentz	\$4,200.00
MK_MAN	Hartstein	\$13,000.00
MK_REP	Fay	\$6,000.00
SA_MAN	Zlotkey	\$10,500.00
SA_REP	Abel	\$11,000.00
	Grant	\$7,000.00
	Taylor	\$8,600.00

Confidential



Synthèse

Ce chapitre vous à permis d'apprendre à :

- utiliser des variables de substitution iSQL*Plus pour stocker temporairement des valeurs
- utiliser des commandes SET pour contrôler l'environnement iSQL*Plus en cours
- utiliser la commande COLUMN pour contrôler l'affichage d'une colonne
- utiliser la commande BREAK pour supprimer les doublons et répartir les lignes en sections
- utiliser les commandes TTITLE et BTITLE pour afficher des en-têtes et des pieds de page

Présentation de l'exercice 7

Dans cet exercice, vous allez :

- créer une interrogation utilisant des variables de substitution pour afficher des valeurs,
- lancer un fichier de commandes contenant des variables.