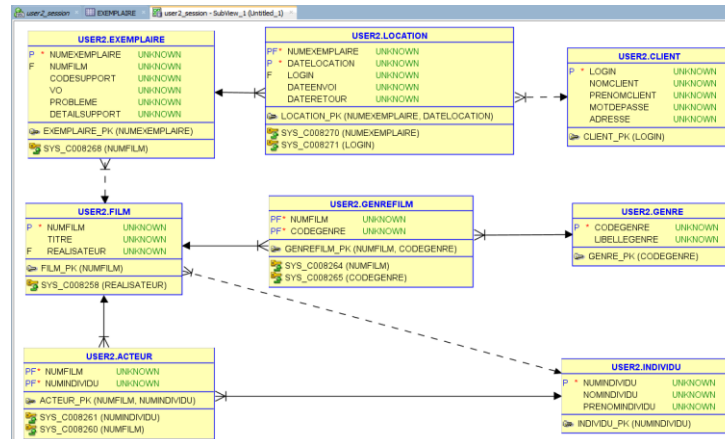


## Série 3 TP 2 – Corrigé



Accéder aux requêtes de création et insertion des données :

<https://livesql.oracle.com/apex/livesql/s/bri01p8koymmrz78qw5etowun>

### Exercice 1 :

Scripts: <https://github.com/DgrinderHZ/TD-CI-GI-FSTE---Answers>

a)

```

Script Output x
Task completed in 0.084 seconds

Hello World

PL/SQL procedure successfully completed.
  
```

b)

```

Script Output x
Task completed in 0.128 seconds

Le nombre de comédies est : 1

PL/SQL procedure successfully completed.
  
```

## Exercice 2 : Exceptions

Une fois les données insérées (3 films) on exécute la première requête qui génère une erreur :

```
Script Output x
Task completed in 0.136 seconds

Error report -
ORA-01422: exact fetch returns more than requested number of rows
ORA-06512: at line 6
01422. 00000 - "exact fetch returns more than requested number of rows"
*Cause: The number specified in exact fetch is less than the rows returned.
*Action: Rewrite the query or change number of rows requested
```

1. Sans gestion d'exception, il faut être sûr que vous avez un film avec 10 exemplaires :

```
Script Output x Query Result x
Task completed in 0.087 seconds

Numéro de film : 1
Titre : The Unique Film
Réalisateur : 1

PL/SQL procedure successfully completed.
```

2. Avec gestion d'exception, maintenant il faut être sûr que vous avez un film avec 10 exemplaires :

Pour voir les scénarios possibles, supprimer/ajouter des données [utiliser ce script](#) pour insérer un deuxième film avec 10 exemplaires.

- 2 Films

NUMEXEMPLAIRE	NUMFILM	CODESUPPORT	VO	PROBLEME	DETAILSUPPORT
1	1	1 DVD	N	(null)	DVD
2	2	1 DVD	N	(null)	DVD
3	3	1 DVD	N	(null)	DVD
4	4	1 DVD	N	(null)	DVD
5	5	1 DVD	N	(null)	DVD
6	6	1 DVD	N	(null)	DVD
7	7	1 DVD	N	(null)	DVD
8	8	1 DVD	N	(null)	DVD
9	9	1 DVD	N	(null)	DVD
10	10	1 DVD	N	(null)	DVD
11	102	6 (null)	(null)	(null)	(null)
12	103	6 (null)	(null)	(null)	(null)
13	104	6 (null)	(null)	(null)	(null)
14	105	6 (null)	(null)	(null)	(null)
15	106	6 (null)	(null)	(null)	(null)
16	107	6 (null)	(null)	(null)	(null)
17	108	6 (null)	(null)	(null)	(null)
18	109	6 (null)	(null)	(null)	(null)
19	110	6 (null)	(null)	(null)	(null)
20	101	6 DVD	N	(null)	DVD

```
Script Output x Query Result x
Task completed in 0.116 seconds

Plusieurs films avec exactement 10 exemplaires trouvés.

PL/SQL procedure successfully completed.
```

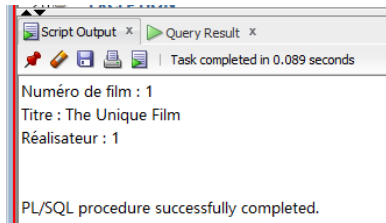
- 1 Film

NUMEXEMPLAIRE	NUMFILM	CODESUPPORT
1	1	1 DVD
2	2	1 DVD
3	3	1 DVD
4	4	1 DVD
5	5	1 DVD
6	6	1 DVD
7	7	1 DVD
8	8	1 DVD
9	9	1 DVD
10	10	1 DVD
11	102	6 (null)
12	103	6 (null)
13	104	6 (null)
14	105	6 (null)
15	106	6 (null)
16	107	6 (null)
17	108	6 (null)
18	109	6 (null)
19	110	6 (null)
20	101	6 DVD

NUMEXEMPLAIRE	NUMFILM	CODESUPPORT	VO	PROBLEME	DETAILSUPPORT
1	1	1 DVD	N	(null)	DVD
2	2	1 DVD	N	(null)	DVD
3	3	1 DVD	N	(null)	DVD
4	4	1 DVD	N	(null)	DVD
5	5	1 DVD	N	(null)	DVD
6	6	1 DVD	N	(null)	DVD
7	7	1 DVD	N	(null)	DVD
8	8	1 DVD	N	(null)	DVD
9	9	1 DVD	N	(null)	DVD
10	10	1 DVD	N	(null)	DVD
11	102	6 (null)	(null)	(null)	(null)
12	103	6 (null)	(null)	(null)	(null)
13	104	6 (null)	(null)	(null)	(null)
14	105	6 (null)	(null)	(null)	(null)
15	106	6 (null)	(null)	(null)	(null)
16	107	6 (null)	(null)	(null)	(null)
17	108	6 (null)	(null)	(null)	(null)
18	109	6 (null)	(null)	(null)	(null)
19	110	6 (null)	(null)	(null)	(null)
20	101	6 DVD	N	(null)	DVD

NUMEXEMPLAIRE	NUMFILM	CODESUPPORT
1	1	1 DVD
2	2	1 DVD
3	3	1 DVD
4	4	1 DVD
5	5	1 DVD
6	6	1 DVD
7	7	1 DVD
8	8	1 DVD
9	9	1 DVD
10	10	1 DVD
11	102	6 (null)

Maintenant exécuter votre code :

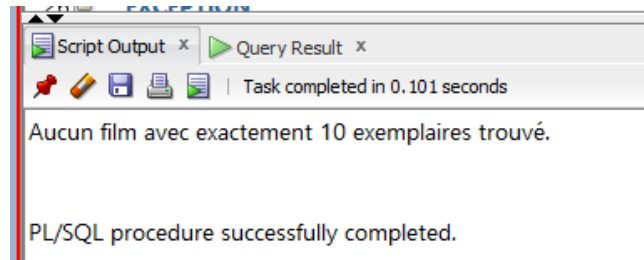


### - 0 Film

Supprimer quelques exemplaires de film avec 10 exemplaires pour avoir la condition

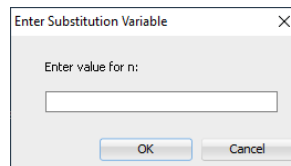
NUMEXEMPLAIRE	NUMFILM	CODESUPPORT	VO	PROBLEME	DETAILSUPPORT
1	1	1 DVD	N	(null)	DVD
2	2	1 DVD	N	(null)	DVD
3	10	1 DVD	N	(null)	DVD
4	102	6 (null)	(null)	(null)	(null)

Maintenant exécuter votre code :

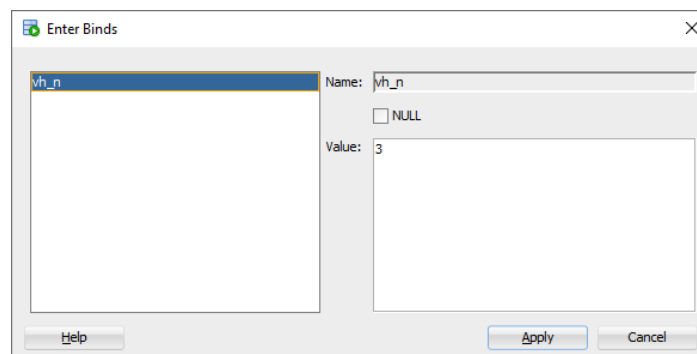


## Exercice 3 :

Scripts: <https://github.com/DgrinderHZ/TD-CI-GI-FSTE---Answers>



Variable hôte (bind variable) exécuter la déclaration et l'initialisation de variable avant votre bloc PLSQL :



## Exercice 4 : Curseurs

Scripts: <https://github.com/DgrinderHZ/TD-CI-GI-FSTE---Answers>

Enter Substitution Variable

Enter value for numIndividu:

1

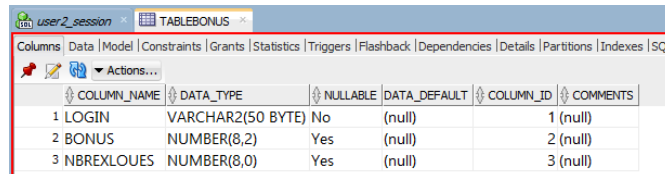
OK Cancel

```
Script Output x Query Result x
Task completed in 2.982 seconds
EXIT WHEN cur%NOTFOUND;
DBMS_OUTPUT.PUT_LINE('Numéro du film : ' || v_numfilm || 'Titre : ' || v_titre || 'Num. Réalisateur: ' || v_realisateur);
END LOOP;
CLOSE cur_films;
END;
Numéro du film : 1Titre: The Unique FilmNum. Réalisateur: 1
PL/SQL procedure successfully completed.
```

## Exercice 5 :

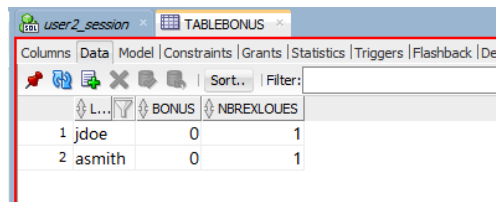
Scripts: <https://github.com/DgrinderHZ/TD-CI-GI-FSTE---Answers>

- a. Création de la table tableBonus



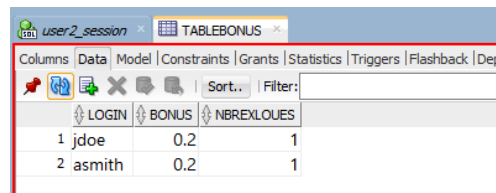
COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 LOGIN	VARCHAR2(50 BYTE)	No	(null)	1 (null)	
2 BONUS	NUMBER(8,2)	Yes	(null)	2 (null)	
3 NBREXLOUES	NUMBER(8,0)	Yes	(null)	3 (null)	

- b. Remplissage de la table tableBonus pour l'année 2023 :



	BONUS	NBREXLOUES
1 jdoe	0	1
2 asmith	0	1

- c. Variante 01: Mise à jour de la table tableBonus avec les bonus selon v\_n1=1 et v\_n2=5



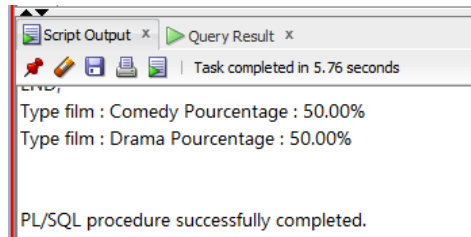
	BONUS	NBREXLOUES
1 jdoe	0.2	1
2 asmith	0.2	1

- d. Affichage des clients avec leur nom, prénom et bonus

```
Script Output x Query Result x
Task completed in 0.094 seconds
Nom : Doe, Prénom : John, Bonus : .2
Nom : Smith, Prénom : Alice, Bonus : .2
PL/SQL procedure successfully completed.
```

## Exercice 6 :

Scripts: <https://github.com/DgrinderHZ/TD-CI-GI-FSTE---Answers>

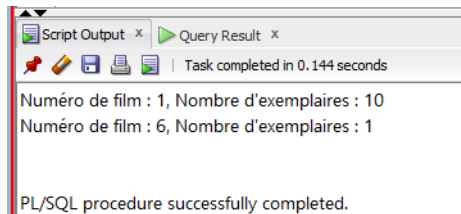


Script Output x Query Result x  
Task completed in 5.76 seconds  
END;  
Type film : Comedy Pourcentage : 50.00%  
Type film : Drama Pourcentage : 50.00%  
PL/SQL procedure successfully completed.

## Exercice 7 :

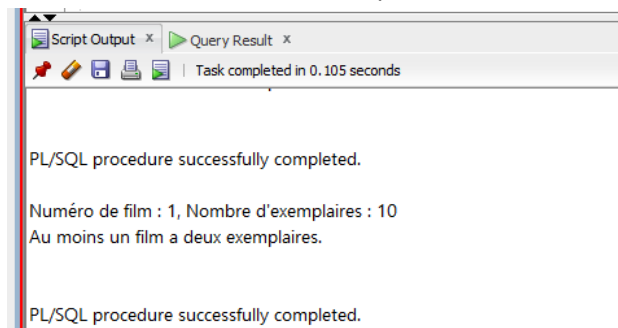
Scripts: <https://github.com/DgrinderHZ/TD-CI-GI-FSTE---Answers>

- a. Ecrire un bloc PL/SQL qui calcule le nombre d'exemplaires pour chaque film



Script Output x Query Result x  
Task completed in 0.144 seconds  
Numéro de film : 1, Nombre d'exemplaires : 10  
Numéro de film : 6, Nombre d'exemplaires : 1  
PL/SQL procedure successfully completed.

- b. Dans ce même bloc, déclarer et utiliser une exception qui permet de quitter le bloc et d'envoyer un message d'erreur si au moins un film a deux exemplaires.



Script Output x Query Result x  
Task completed in 0.105 seconds  
PL/SQL procedure successfully completed.  
Numéro de film : 1, Nombre d'exemplaires : 10  
Au moins un film a deux exemplaires.  
PL/SQL procedure successfully completed.