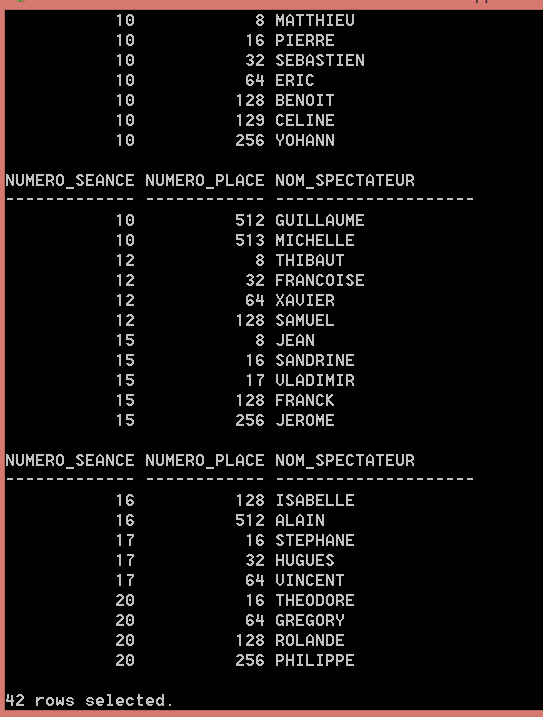
BASE DE DONNÉES – TD5 (TRANSACTIONS)

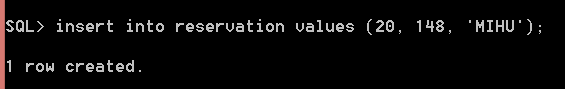
**LÊ NGUYỄN MINH HƯƠNG**

# 1. Effectuez successivement les opérations suivantes.

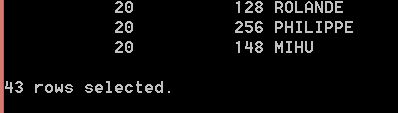
## (a) Affichez toutes les réservations.



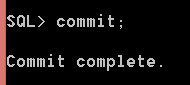
## (b) Insérez une réservation.



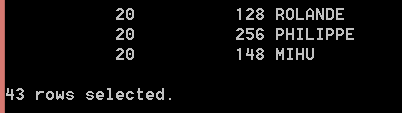
## (c) Affichez toutes les réservations.



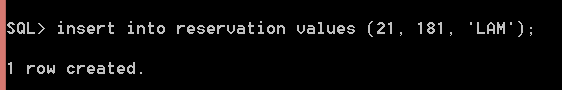
## (d) Tapez l’instruction commit; .



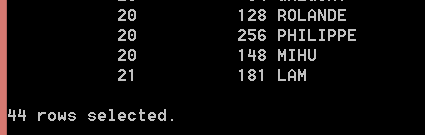
## (e) Affichez toutes les réservations.



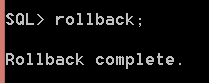
## (f) Insérez une nouvelle réservation.



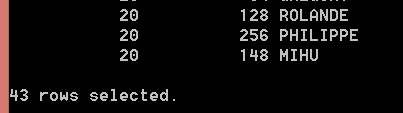
## (g) Affichez toutes les réservations.



## (h) Tapez l’instruction rollback; .



## (i) Affichez toutes les réservations.



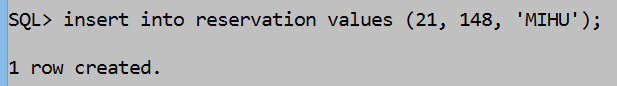
## (j) Concluez !

- COMMIT : enregistrer les nouveaux changements dans la base

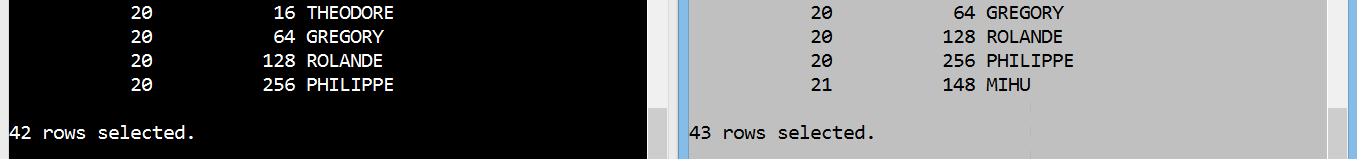
- ROLLBACK : enlever tous les nouveaux changements dans la base, réalisés par une seule session, si ceux-ci n'ont pas été admis par l'instruction COMMIT

# 2. Lancez une seconde session sqlplus et procédez comme suit.

## (a) Insérez une réservation dans cette seconde session.

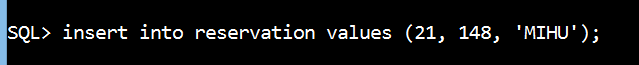


## (b) Affichez cette même réservation pour chacune des deux sessions. Commentez !



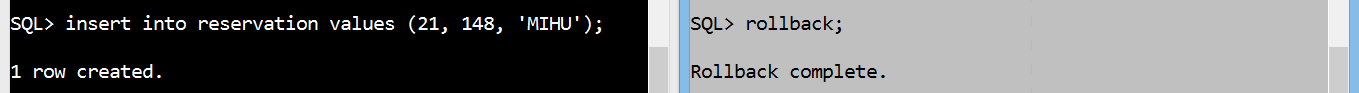
La nouvelle réservation n’est pas ajoutée dans la base de la 1e session.

## (c) Insérez cette même réservation dans la première session. Que se passe-t-il dans cette première session ?



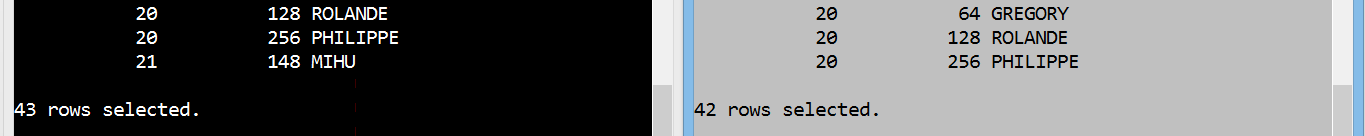
Le programme ne marche pas. Il attend un ROLLBACK / COMMIT.

## (d) Effectuez un rollback dans la seconde session.



Quand on effectue un ROLLBACK dans la 2e session, la 1e peut désormais ajouter la nouvelle réservation.

## (e) Affichez cette même réservation pour chacune des deux sessions.

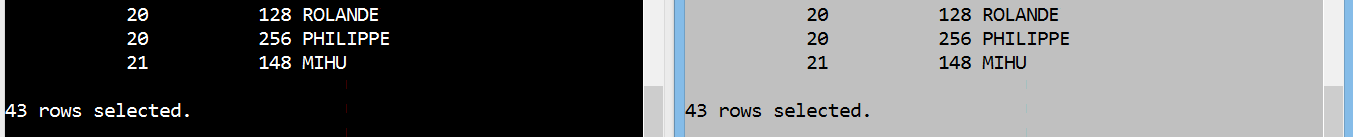


La nouvelle réservation dans le 2e session est supprimée ( en raison de l’instruction ROLLBACK )

## (f) Effectuez un commit dans la première session.

La nouvelle réservation sera mise à jour dans toute la base, dans toutes les sessions.

## (g) Affichez cette même réservation pour chacune des deux sessions.



## (h) Concluez !

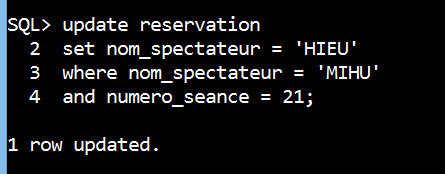
Il faut absolument exécuter l'instruction COMMIT après chaque nouvelle modification dans la base, sinon les autres sessions de la base ne remarquent pas ces changements ( elles attendent pour un COMMIT afin de pouvoir insérer de nouvelles données ).

# 3. Effectuez successivement les opérations suivantes.

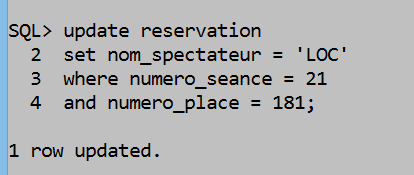
( données R1 : 21, 148, ‘MIHU’ )

( données R2 : 21, 181, ‘LAM’ )

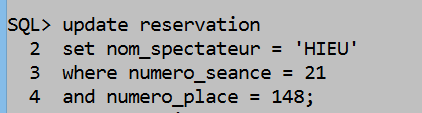
## (a) Modifiez une réservation R1 dans la première session.



## (b) Modifiez une réservation R2 dans la seconde session.

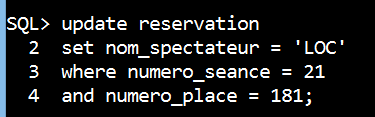


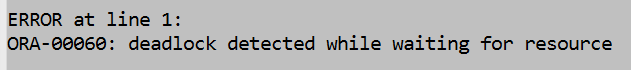
## (c) Modifiez la réservation R1 dans la seconde session.



Le programme ne marche pas. Il attend un COMMIT.

## (d) Modifiez la réservation R2 dans la première session.





## (e) Effectuez un rollback dans la première session.

## (f) Effectuez un rollback dans la seconde session.

## (g) Concluez !

DEALOCK se produit, car la 2e session attend un COMMIT à partir de la 1e session, mais au contraire la 1e session attend un COMMIT à partir de la 2e session, alors ces deux sessions tombes dans une boucle 🡪 DEADLOCK ;

## (h) Fermez la seconde session sqlplus.