

# L3 informatique - Implémentation des SGBDs

## Transactions : TP1

8 décembre 2019

### 1 Exercice 1

1. Ouvrez deux sessions, Session 1 et Session 2, sous le même login.
2. Exécutez les commandes suivantes à partir de la Session 1 :

```
INSERT INTO EMP VALUES
(7000,'Petit Lion', 'SALESMAN',7902,to_date('17-12-1980','dd-mm-yyyy'),800,NULL,20);
INSERT INTO EMP
VALUES (7001,'Chaussette','SALESMAN',7698,to_date('20-2-1981','dd-mm-yyyy'),1600,300,20);
```

3. Retournez à la Session 2 et vérifiez si vos modifications sont visibles à partir de cette session. Que constatez-vous ?
4. A partir de la Session 1, faites un COMMIT des modifications et vérifiez si les modifications sont connues dans la Session 2.
5. A partir de la Session 1, augmenter le salaire de l'employé 'petit lion' de 200 Euros.
6. A partir de la Session 2, mettez le salaire de l'employé 'petit lion' à 700 Euros. Que se passe-t-il ?
7. Faites un COMMIT dans la Session 1. Que se passe-t-il ? Faites un select dans les 2 sessions pour voir la modification.
8. Faites un COMMIT dans la deuxième session. Faites un select dans les 2 sessions pour voir la modification.
9. Utilisez un SELECT FOR UPDATE sur la Session 1 et essayez de modifier le salaire des employés à partir de la Session 2.

### 2 Exercice 2

#### Rappel : mode de fonctionnement par défaut d'Oracle

- Les lectures ne bloquent ni les autres lectures ni les écritures.
  - Les lectures ne sont bloquées par rien, même pas par un blocage d'une table en mode exclusif.
  - Il n'y a pas de lecture impropre.
  - Il n'y a pas de pertes de mises à jour.
  - Il peut y avoir des lectures non reproductibles.
  - Il peut y avoir des lignes fantômes.
1. Pour chacun des scénarios ci-dessous, exécuter la séquence d'instructions à partir de deux sessions (session 1 et session 2) et expliquer quels sont les phénomènes observés ?

#### (a) Scénario 1

```
Session 1 : select sal from emp where empno = 7369;
Session 2 : update emp set sal= 802 where empno = 7369;
Session 2 : commit;
Session 1 : select sal from emp where empno = 7369;
```

#### (b) Scénario 2

```
Session 1 : update emp set sal= 801 where empno = 7369;
Session 2 : select sal from emp where empno = 7369;
Session 1 : commit;
Session 2 : select sal from emp where empno = 7369;
```

(c) **Scénario 3**

**Session 1 :** update emp set sal= 803 where empno = 7369 ;  
**Session 2 :** update emp set sal= 804 where empno = 7369 ;  
**Session 1 :** commit ;  
**Session 2 :** commit ;

(d) **Scénario 4**

**Session 1 :** update emp set sal= 805 where empno = 7369 ;  
**Session 2 :** update emp set sal = 1300 where empno=7521 ;  
**Session 1 :** update emp set sal = 1300 where empno=7521 ;  
**Session 2 :** update emp set sal= 805 where empno = 7369 ;

attendre un moment et observer ce qui se passe.

**Session 1 :** ROLLBACK ;

Expliquer ce qui s'est passé.

### 3 Exercice 3

#### A. Empêcher les lectures non reproductibles

Que pouvez-vous faire pour empêcher les lectures non reproductibles, dans chacun des cas suivants :

- dans le cas où la transaction ne modifie aucune donnée.
- dans le cas où elle modifie des données.

#### B. Empêcher les lignes fantômes

Mêmes questions que l'exercice précédent, mais pour les lignes fantômes.

— **Read only**

1. Ouvrez une nouvelle transaction en "READ ONLY".
2. Ouvrez en parallèle une deuxième transaction dans laquelle vous modifiez les salaires des employés.
3. Validez cette deuxième transaction. Voyez-vous les modifications dans la première transaction ?
4. Essayez de modifier des données dans la première transaction.
5. Que se serait-il passé si la première transaction n'avait pas été en "READ ONLY" ? Vérifiez-le après avoir terminé la première transaction.

— **Mode serialisable**

Essayez de faire cet exercice en devinant ce qui va se passer avant de lancer chaque commande.

1. Ouvrez 2 sessions de travail avec des transactions T1 et T2.
2. Passez T1 en mode sérialisé.
3. T1 affiche tous les noms et salaires des employés.
4. T2 modifie le salaire de l'employé 'WARD'.
5. T1 affiche à nouveau tous les salaires. Quel salaire voit-il pour l'employé 'WARD' ? Pourquoi ?
6. T2 valide sa transaction.
7. T1 affiche à nouveau tous les salaires. Quel salaire voit-il pour l'employé 'WARD' ? Pourquoi ?
8. T1 modifie le salaire de l'employé 'WARD'. Que se passe-t-il ? Pourquoi ?
9. T1 modifie le salaire d'un autre employé. Que se passe-t-il ?

Voyez-vous une différence avec le mode par défaut d'Oracle ? Explications ?