

# Manipulation d'une base de données PostgreSQL avec les transactions longues Corrigé

A partir de la base tp et de la relation 'batiments' :

Demandez l'autorisation d'effectuer des MAJ et suppressions sur la relation 'batiments'

```
-- premiere etape Positionner les transactions longues  
Select EnableLongTransactions();
```

```
-- creation du trigger  
Select CheckAuth('batiments', 'id');
```

Verrouillez le bâtiment '1' pour l'utilisateur 'michel'

```
-- retour positif  
Select LockRow('batiments','1','michel');
```

Montrez que personne d'autres que 'michel' ne peut manipuler le bâtiment '1'

```
-- retour negatif  
Select LockRow('batiments','1','joe');
```

Verrouillez le bâtiment '2' pour l'utilisateur 'joe'

```
-- retour positif  
Select LockRow('batiments','2','joe');
```

Essayez de modifier le bâtiment '2' en étant l'utilisateur 'michel'

```
begin transaction;  
-- retour positif  
Select AddAuth('michel');  
  
-- retour negatif  
update batiments set nom = 'Refuse' where id = 2;  
  
rollback;
```

Montrez que le bâtiment '1' est une 'Mairie'

```
select nom from batiments where id = 1;
```

Modifiez le bâtiment '1' en étant l'utilisateur 'michel' : le nom devient 'Ecole'

```
begin transaction;
```

```
Select AddAuth('michel');
```

```
-- retour positif
```

```
update batiments set nom = 'Ecole' where id = 1;
```

Montrez que la modification est faite

```
select nom from batiments where id = 1;
```

Restaurez l'état initial (au choix)

```
a) -- retour positif
```

```
update batiments set nom = 'Mairie' where id = 1;
```

```
select nom from batiments where id = 1;
```

```
commit;
```

```
b) rollback ;
```

```
Select UnlockRows('michel');
```

```
Select UnlockRows('joe');
```

```
Select DisableLongTransactions();
```

```
\q
```

Dans le cadre de deux sessions de travail

Session 1 :

Posez un verrou en étant l'utilisateur 'michel' sur le bâtiment '1'

```
-- premiere etape Positionner les transactions longues
```

```
Select EnableLongTransactions();
```

```
-- retour positif
```

```
Select LockRow('batiments','1','michel');
```

Posez un verrou en étant l'utilisateur 'joe' sur le bâtiment '2'

```
-- retour positif
```

```
Select LockRow('batiments','2','joe');
```

Terminez la session de travail

```
\q
```

Session 2 :

Première transaction :

Modifiez le bâtiment '1' en étant 'michel' le nom devient 'Ecole'

-- première étape Positionner les transactions longues

```
Select EnableLongTransactions();
```

-- michel modifie le bâtiment 1

```
begin transaction;
```

```
Select AddAuth('michel');
```

```
select nom from batiments where id = 1;
```

-- retour positif

```
update batiments set nom = 'Ecole' where id = 1;
```

```
select nom from batiments where id = 1;
```

```
commit;
```

Deuxième transaction :

Montrez que 'joe' ne peut pas modifier le bâtiment '2' sans autorisation

-- joe ne peut pas modifier sans autorisation

```
begin transaction;
```

```
update batiments set nom = 'Mairie' where id = 2;
```

```
rollback;
```

Restaurez l'état initial

-- restauration du contexte

```
begin transaction;
```

```
Select AddAuth('michel');
```

```
update batiments set nom = 'Mairie' where id = 1;
```

```
commit;
```

```
Select UnlockRows('michel');
```

```
Select UnlockRows('joe');
```

```
Select DisableLongTransactions();
```

\q