

```
/******  
/****** Jeu de test *****/
```

Les commandes que nous donnerons sont syntaxiquement correctes mais toutes ne sont pas sémantiquement correctes.

Pour les commandes qui ne sont pas sémantiquement correctes le but de montrer comment l'interpréteur réagit face à des requêtes autrement dit ces requêtes vont nous permettre de présenter tous les contrôles qui ont été effectués.

**NB : Les requêtes sémantiquement incorrectes sont en orange.**

Aussi, il est possible d'écrire dans un fichier un lot de requêtes qu'on pourra exécutés simultanément en faisant passer le fichier en paramètre. Un fichier nommé fich\_param existe, on pourra le tester en tapant :

```
invite@loue-laptop:~/Bureau/ok/projet$ ./a.out<fich_param  
*****  
* INTERPRETEUR DE COMMANDE shellSQL *  
* Version: 1.1 *  
* Copyright LOUE Boureima Arnaud et OUEDRAOGO Pascal *  
* All right reserved *  
*****  
shellSQL > la table test2 a ete creee avec succes !!  
shellSQL > la table test3 a ete creee avec succes !!  
shellSQL > 1 ligne inserée !!  
shellSQL > 1 ligne inserée !!  
shellSQL > 1 ligne inserée !!  
shellSQL > 1 ligne inserée !!  
shellSQL > invite@loue-laptop:~/Bureau/ok/projet$
```

## Les commandes

### 1. CREATE

- create table essai0(id integer primary key,nom text,id\_parti text foreign key references parti(code));
- create table essai0(id integer primary key,nom text primary key);
- create table essai0(id integer primary key,nom text,id\_parti text foreign key references parti(id));

### 2. INSERT

- Insert into test values(7,'00E');

- insert into essai0 values(1,'TOTO','000A');
- insert into essai0 values(1,'TATI','000A');
- insert into compte values(5,-2000.00);
- insert into compte(6,NULL);
- insert into compte(NULL,25000.00);
- insert into etu values(8,'GOHOU','Michel',3,67,2);
- insert into compte values(7);

### 3. SELECT

- select id,solde from compte where solde<0;
- select \* from compte where solde!=0.0;
- select depute.nom,parti.nom,region.nom from depute,parti,region where depute.id\_parti=parti.id and depute.id\_region=region.id ;
- select depute.nom,depute.id\_parti,parti.id,parti.nom from depute join parti where depute.id\_parti=parti.id ;
- select \* from etu where nom='LOUE' ;
- select \* from etu where nom<'LOUE';

### 4. UPDATE

- update footballeur set nom='PEDRO RODRIGEZ' where code='00G';
- update compte set solde=0 where solde<0;
- update compte set id=3 where sole>0;

### 5. DELETE

- delete from footballeur where nom='Ronaldo' ;
- delete from compte where solde>0;
- delete from footballeur where nom='Raoul';
- delete from compte where attribut=26000;

### 6. ALTER

- alter table depute add ethnie text ;
- alter table depute drop ethnie ;
- alter table depute add id\_ethnie text foreign key references Ethnie(id) ;
- alter table parti drop id ;

### 7. RENAME

- rename essai0 to essai ;
- rename client to clientele ;

### 8. DROP

- drop table essai ;
- drop client ;

## 9. DESC

- desc depute ;
- desc client ;