

---

## MISE EN PLACE

---

### Installer la base de données

---

1. Lancez la console d'administration du SGBD PostGres  
<http://a-pedagoarles-postgresql.aix.univ-amu.fr/phppgadmin/> ) et connectez-vous avec vos paramètres;
2. Récupérez le script de création des tables (***Postgres-Coop.sql***) sur ma plateforme et copier/coller son contenu dans la fenêtre SQL.
3. Lancer le script.
4. Vos tables sont "normalement" créées et renseignées dans le schéma de données ***ctrl3\_cooperative***

## PRESENTATION

Le base de données permet la gestion des silos d'une coopérative agricole.

**Une céréale peut être stocké dans un ou plusieurs silos**, un silo ne peut contenir qu'en seule céréale (pas de mélange de variété possible) sa capacité maximum est définie dans la colonne *qtemax* de la table *silo*, la quantité du stock courant pour un silo est donnée dans le colonne *qtestock*.

codesilo	codecereale	qtestock	qtemax
D	MS440	608	700
E	MS440	48	500

Un contrat permet de satisfaire un acheteur, il est traité par un négociant, il concerne une seule céréale.

Un contrat peut être livré en plusieurs fois, le total des quantités livrées (*qteliv*) pour un contrat ne peut dépasser la quantité indiquée sur le contrat (*qtecd*)

nocontrat	dateliv	qteliv	qtecd	
1477	2002-01-20 00:00:00	45	70	← Livraison complète du contrat
1477	2002-01-22 00:00:00	25	70	
1478	2002-02-20 00:00:00	10	45	← Livraison partielle du contrat
1478	2002-02-28 00:00:00	10	45	
1478	2002-03-03 00:00:00	10	45	
1479	2002-02-25 00:00:00	50	50	← Livraison complète du contrat
1480	2002-02-22 00:00:00	25	50	← Livraison complète du contrat
1480	2002-03-30 00:00:00	25	50	
1481	2002-03-12 00:00:00	80	200	← Livraison partielle du contrat
1481	2002-03-25 00:00:00	80	200	

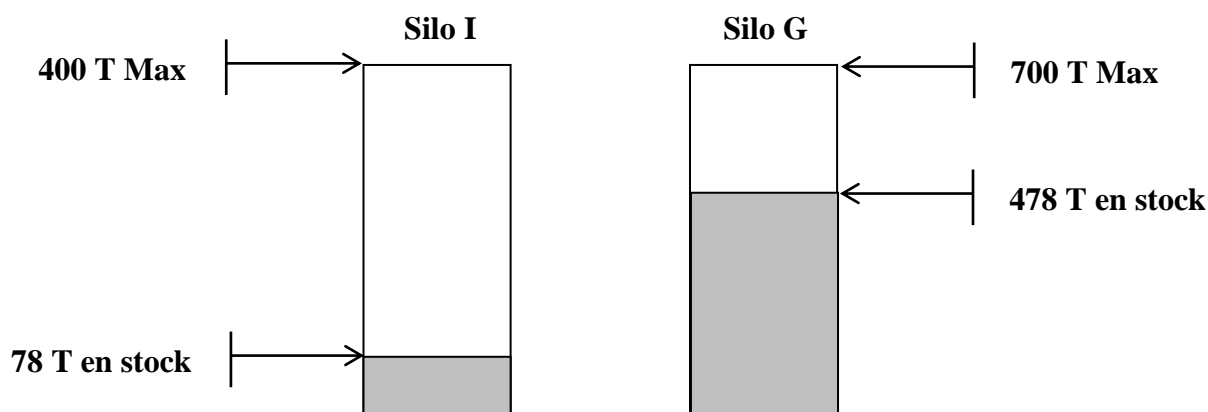
Image de la table *livrer* jointe avec la table *contrat* (il n'y a que la colonne *qtecd* affichée de la table *contrat*)

### Principe de remplissage d'un silo

Pour remplir les silos d'une céréale, il faut liste les silos qui contiennent la céréale à stocker et ensuite commencer par remplir le silo le plus plein.

Exemple : on désire stocker 600 tonnes de céréales Colza NTR 120 (CZ120),

- Il y a deux silos qui stockent cette céréale, le I et le G
- Le G est le plus plein, on commencera donc par remplir celui-ci, de 700-478 soit 222 tonnes, après cette opération le silo G sera plein,
- Il restera, 600-222 soit 378 tonnes de céréale (CZ120) à stocker qui seront stockées dans le silo I,
- On remplit ensuite le silo I de 400-78 soit 322 tonnes, après cette opération le silo I sera plein,
- Il reste 378-322 soit 56 tonnes qui ne sont pas stockées



**REMARQUES IMPORTANTES**

**Respecter les noms que je vous donne pour les fonctions et triggers.  
Me fournir les requêtes qui permettent de tester les fonctions et triggers.**

**EXERCICE 1 : LES FONCTIONS**

1. Ecrire la fonction ***stock\_cereale(refCereale)*** qui retourne la quantité du stock d'une céréale passée en paramètre. La céréale doit exister dans la BDD, message d'erreur si la céréale n'existe pas.

**Rappel** : une céréale peut être stockée dans plusieurs silos.

**Paramètre** : refCereale : référence de la céréale ;

**Vous utiliserez les quatre valeurs de paramètre suivantes pour faire vos tests:**

**('MS440') , ('CZ120') , ('BL500') / ('BE500')**

2. Ecrire la fonction ***verif\_contrat(refCereale,qte)*** qui permet de vérifier si le stock total d'une céréale (***refCereale***) permet de satisfaire un contrat (une commande) pour une quantité donnée(***qte***).

Si la quantité en stock est suffisante la fonction doit retourner le booléen TRUE sinon FALSE.  
La céréale doit exister dans la BDD, message d'erreur si la céréale n'existe pas.

**Rappel** : une céréale peut être stockée dans plusieurs silos.

**Paramètre** : refCereale : référence céréale ;

qte : Quantité demandée ;

**Vous utiliserez les trois couples de valeurs suivants pour faire vos tests :**

**('MS440',700) , ('BL500',45) , ('BE500',980)**

3. Ecrire la fonction ***ajout\_cereale(refCereale,qte)*** qui permet d'ajouter une céréale dans les silos. Cette fonction cherchera les silos qui contiennent cette céréale et remplira le silo le plus plein et complètera le stockage dans les autres silos toujours en complétant les silos les plus pleins. La fonction retourne la quantité de céréale qui n'a pas pu être stockée. La céréale doit exister dans la BDD, message d'erreur si la céréale n'existe pas

**ATTENTION :**

- Cet exercice demande une bonne dose de réflexion,
- L'utilisation d'un curseur qui retourne les silos qui stockent la céréale à ajouter peut s'avérer utile,
- **Je vous conseille de réaliser cet exercice en dernier .....**

**Rappel** : une céréale peut être stockée dans plusieurs silos.

**Paramètre** : refCereale : référence céréale ;

qte : Quantité à ajouter ;

**Vous utiliserez les cinq couples de valeurs suivants pour faire vos tests :**

**('MS440',92) , ('MS440',300) , ('MS440',200) , ('MS440',200) ,  
('MY440',200)**

**EXERCICE 2 : LES TRIGGERS**

1. Ecrire le trigger **trig\_insert\_contrat()** qui lors de l'insertion d'un contrat vérifie :
  - Que la céréale existe, message d'erreur le cas contraire
  - Que le négociant existe, message d'erreur le cas contraire
  - Que la quantité de céréale demandée est supérieur à 0, message d'erreur le cas contraire
  - Que la quantité de céréale demandée peut être satisfaite (il y en a assez en stock), message d'erreur le cas contraire

**ATTENTION :** Lors de l'insertion la date du contrat n'est pas renseignée ni le numéro du contrat, le trigger doit valoriser cette date à la date courante (fonction `current_timestamp`) et générer un numéro de contrat valide, avant l'insertion effective.

**!!!!!! Ce contrat n'est pas encore livré (puisque'il est nouveau) il ne faut donc pas modifier la quantité en stock de céréale !!!!!**

***Vous utiliserez les jeux de tests suivants pour tester votre fonction***

```
INSERT INTO contrat (codecereale,no_negociant,qtecde) VALUES('CA120',48,400);
INSERT INTO contrat (codecereale,no_negociant,qtecde) VALUES('CZ120',80,400);
INSERT INTO contrat (codecereale,no_negociant,qtecde) VALUES('CZ120',48,800);
INSERT INTO contrat (codecereale,no_negociant,qtecde) VALUES('CZ120',48,400);
```

**ANNEXE 1 : MPD DE LA GESTION DES SILOS**