

SAE 3.01 Partie WEB :

Triggers et Procédures

SOMMAIRE

Triggers :	2
Procédures :	6

Triggers :

Nom du déclencheur	<input type="text" value="insert_stock_produit"/>
Table	<input type="text" value="Produit"/>
Moment	<input type="text" value="BEFORE"/>
Évènement	<input type="text" value="INSERT"/>


```
1 IF NEW.stock >= stockMax THEN
2     SIGNAL SQLSTATE '45000'
3     SET MESSAGE_TEXT = 'La valeur de stock doit être
inférieure à stockMax';
4 END IF
```

INSERT INTO Produit (numProduit, coutAchat, prixVente, codeBarre, referenceInterne, nomProduit, seuilReapprovisionnement, stockMax, stock, fraisSupplementaires, numCategorie, codeType, numRegroupement, numProduitCompose, section, numSousCategorie, numCategorieInferieur)
VALUES (60, 10, 20, '1234567890', 'ref123', 'ProduitTest', 5, 50, 55, 5, 2, 1, NULL, NULL, 'new', 1, 1);

Erreur

Requête SQL : [Copier](#)

```
INSERT INTO Produit (numProduit, coutAchat, prixVente, codeBarre, referenceInterne, nomProduit, seuilReapprovisionnement, stockMax, stock, fraisSupplementaires, numCategorie, codeType, numRegroupement, numProduitCompose, section, numSousCategorie, numCategorieInferieur)
VALUES (60, 10, 20, '1234567890', 'ref123', 'ProduitTest', 5, 50, 55, 5, 2, 1, NULL, NULL, 'new', 1, 1);
```

MySQL a répondu : 

#1644 - La valeur de stock doit être inférieure à stockMax

Nom du déclencheur	<input type="text" value="update_stock_produit"/>
Table	<input type="text" value="Produit"/>
Moment	<input type="text" value="BEFORE"/>
Évènement	<input type="text" value="UPDATE"/>

```
1 IF NEW.stock > OLD.stockMax THEN
2     SIGNAL SQLSTATE '45000'
3     SET MESSAGE_TEXT = 'La valeur de stock doit être
inférieure à stockMax';
4 END IF
```


```
1 UPDATE Produit
2 SET stock = 70
3 WHERE numProduit = 1;
```

UPDATE Produit

SET stock = 70

WHERE numProduit = 1;

```
UPDATE Produit
SET stock = 70
WHERE numProduit = 1;
```

MySQL a répondu : 

#1644 - La valeur de stock doit être inférieure à stockMax

Nom du déclencheur	<u>Contrainte_qteCmd_stock</u>
Table	LigneCde
Moment	BEFORE
Évènement	INSERT
Définition	<pre> 1 BEGIN 2 DECLARE stockP INT; 3 4 SELECT stock INTO stockP 5 FROM Produit 6 WHERE numProduit = NEW.numProduit; 7 8 9 IF NEW.quantiteCommandee > stockP THEN 10 SIGNAL SQLSTATE '45000' 11 SET MESSAGE_TEXT = 'La quantité commandée ne peut pas dépasser la quantité en stock'; 12 END IF; 13 END </pre>

```

1 INSERT INTO LigneCde (numCommande, numProduit, quantiteCommandee, quantiteLivree)
2 VALUES (2, 1, 24, 24);

```

INSERT INTO LigneCde (numCommande, numProduit, quantiteCommandee,
quantiteLivree)
VALUES (2, 1, 24, 24);

```

INSERT INTO LigneCde (numCommande, numProduit, quantiteCommandee, quantiteLivree)
VALUES (2, 1, 24, 24);

```

MySQL a répondu :

#1644 - La quantité commandée ne peut pas dépasser la quantité en stock

Nom du déclencheur: MajStock

Table: LigneCde

Moment: AFTER

Évènement: INSERT

```

1 UPDATE Produit
2   SET stock = stock - (SELECT quantiteCommandee
3                       FROM LigneCde
4                       WHERE numProduit = NEW.numProduit
5                       AND numCommande = NEW.numCommande)
6   WHERE numProduit = NEW.numProduit

```

+ Options

	numProduit	coutAchat	prixVente	codeBarre	referenceInterne	nomProduit	seuilReapprovisionnement	stock	stockMax	fraisSup
<input type="checkbox"/> Éditer <input type="checkbox"/> Copier <input type="checkbox"/> Supprimer	1	200	299	1234567890	REF001	Guitare Acoustique	10	23	50	5

On voit qu'il y a 23 produits en stock

On fait donc le test suivant :

INSERT INTO LigneCde (numCommande, numProduit, quantiteCommandee, quantiteLivree)

VALUES (2, 1, 22, 22);

```

1 INSERT INTO LigneCde (numCommande, numProduit, quantiteCommandee, quantiteLivree)
2 VALUES (2, 1, 22, 22);
3

```

✓ 1 ligne insérée. (traitement en 0.0982 seconde(s).)

```

INSERT INTO LigneCde (numCommande, numProduit, quantiteCommandee, quantiteLivree) VALUES (2, 1, 22, 22);

```

+ Options

	numProduit	coutAchat	prixVente	codeBarre	referenceInterne	nomProduit	seuilReapprovisionnement	stock	stockMax	fraisSup
<input type="checkbox"/> Éditer <input type="checkbox"/> Copier <input type="checkbox"/> Supprimer	1	200	299	1234567890	REF001	Guitare Acoustique	10	1	50	5

Il n'y a bien plus qu'1 produit en stock maintenant !

Procédures :

Nom de la procédure	PasserCommande				
Type	PROCEDURE				
	Direction	Nom	Type	Taille/Valeurs*	Options
Paramètres	IN	p_numCommande	DECIMAL	15	Supprimer
	IN	p_numClient	INT	15	Supprimer
	IN	p_numProduit	DECIMAL	15	Supprimer
	IN	p_quantiteCommande	DECIMAL	22	Supprimer
	IN	p_montantFrais	DECIMAL	15	Supprimer
	IN	p_montant	DECIMAL	15	Supprimer

```

1 BEGIN
2   DECLARE v_stock DECIMAL(10);
3
4   SELECT stock INTO v_stock
5   FROM Produit
6   WHERE numProduit = p_numProduit;
7
8   IF v_stock IS NULL THEN
9     SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Il n'y a plus ce produit en stock';
10  ELSE
11    IF p_quantiteCommandee <= 0 THEN
12      SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Il faut commander au moins un produit';
13    ELSE
14      IF p_quantiteCommandee > v_stock THEN
15        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Pas assez de produits en stock. Veuillez commander
moins de produit';

```

```

16    ELSE
17      INSERT INTO Commande (numCommande, dateCommande, montantFrais, numClient, montant)
18      VALUES (p_numCommande, NOW(), p_montantFrais, p_numClient, p_montant);
19
20      INSERT INTO LigneCde (numProduit, numCommande, quantiteCommandee)
21      VALUES (p_numProduit, p_numCommande, p_quantiteCommandee);
22    END IF;
23  END IF;
24 END IF;
25 END

```

		numProduit	coutAchat	prixVente	codeBarre	referenceInterne	nomProduit	seuilReapprovisionnement	stock	stockMax	fre
<input type="checkbox"/>	Éditer	1	200	299	1234567890	REF001	Guitare Acoustique	10	1	50	5
<input type="checkbox"/>	Éditer	2	150	249	0987654321	REF002	Piano Numérique	5	20	20	8

Ici on voit qu'il y a 20 produits en stock pour le Piano Numérique

```
1 SET @numCommande = 202;  
2 SET @numClient = 1;  
3 SET @numProduit = 2;  
4 SET @quantiteCommandee = 12;  
5 SET @montantFrais = 10.00;  
6 SET @montant = 50.00;  
7  
8 CALL PasserCommande(  
9     @numCommande,  
10    @numClient,  
11    @numProduit,  
12    @quantiteCommandee,  
13    @montantFrais,  
14    @montant  
15 );  
16  
17 SELECT *  
18 FROM Commande  
19 WHERE numCommande = @numCommande;  
20  
21 SELECT *  
22 FROM LigneCde  
23 WHERE numCommande = @numCommande;  
24
```

```
SET @numCommande = 202;  
SET @numClient = 1;  
SET @numProduit = 2;  
SET @quantiteCommandee = 12;  
SET @montantFrais = 10.00;  
SET @montant = 50.00;
```

```
CALL PasserCommande(  
    @numCommande,  
    @numClient,  
    @numProduit,  
    @quantiteCommandee,  
    @montantFrais,  
    @montant  
);
```

```
SELECT *  
FROM Commande  
WHERE numCommande = @numCommande;
```

```
SELECT *  
FROM LigneCde
```

WHERE numCommande = @numCommande;

Attacher la zone SQL

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0010 seconde(s).)

`SET @numCommande = 202;`

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0011 seconde(s).)

`SET @numClient = 1;`

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0035 seconde(s).)

`SET @numProduit = 2;`

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0014 seconde(s).)

`SET @quantiteCommandee = 12;`

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0010 seconde(s).)

`SET @montantFrais = 10.00;`

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0011 seconde(s).)

`SET @montant = 50.00;`

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0573 seconde(s).)

`CALL PasserCommande(@numCommande, @numClient, @numProduit, @quantiteCommandee, @montantFrais, @montant);`

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0.0034 seconde(s).)

`SELECT * FROM Commande WHERE numCommande = @numCommande;`

☐ Profilage
 [\[Éditer en ligne \]](#)
[\[Éditer \]](#)
[\[Expliquer SQL \]](#)
[\[Créer le code source PHP \]](#)
[\[Actualiser \]](#)

☐ Tout afficher
 Nombre de lignes : 25
 Filtrer les lignes:

+ Options

	numCommande	dateCommande	montantFrais	montant	numClient	idDescription
<input type="checkbox"/> Éditer <input type="checkbox"/> Copier <input type="checkbox"/> Supprimer	202	2023-12-17	10	50	1	NULL

☐ Tout cocher
 Avec la sélection :
 ☐ Éditer
 ☐ Copier
 ☐ Supprimer
 ☐ Exporter

☐ Tout afficher
 Nombre de lignes : 25
 Filtrer les lignes:

Opérations sur les résultats de la requête

☐ Imprimer
 ☐ Copier dans le presse-papiers
 ☐ Exporter
 ☐ Afficher le graphique
 ☐ Créer une vue

☐ Conserver cette requête SQL dans les signets

Intitulé :

☐ Signet visible pour les autres utilisateurs

Conserver cette requête SQL dans les signets

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0.0033 seconde(s).)

`SELECT * FROM LigneCde WHERE numCommande = @numCommande;`

☐ Profilage
 [\[Éditer en ligne \]](#)
[\[Éditer \]](#)
[\[Expliquer SQL \]](#)
[\[Créer le code source PHP \]](#)
[\[Actualiser \]](#)

☐ Tout afficher
 Nombre de lignes : 25
 Filtrer les lignes:

+ Options

	numProduit	numCommande	quantiteCommandee	quantiteLivree
<input type="checkbox"/> Éditer <input type="checkbox"/> Copier <input type="checkbox"/> Supprimer	2	202	12	NULL

☐ Tout cocher
 Avec la sélection :
 ☐ Éditer
 ☐ Copier
 ☐ Supprimer
 ☐ Exporter

☐ Tout afficher
 Nombre de lignes : 25
 Filtrer les lignes:

+ Options

	numProduit	coutAchat	prixVente	codeBarre	referenceInterne	nomProduit	seuilReapprovisionnement	stock	stockMax	fraisSu
<input type="checkbox"/> Éditer <input type="checkbox"/> Copier <input type="checkbox"/> Supprimer	1	200	299	1234567890	REF001	Guitare Acoustique	10	1	50	5
<input type="checkbox"/> Éditer <input type="checkbox"/> Copier <input type="checkbox"/> Supprimer	2	150	249	0987654321	REF002	Piano Numérique	5	8	20	8

On voit qu'il n'y a maintenant bien plus que 8 produits en stock.

J'ai choisie de ne pas traiter l'idDescription et la quantiteLivree dans ma procédure. Pour les rajouter il suffirait juste de rajouter 2 paramètres IN p_idDescription et p_quantiteLivree et de les insert lors des insertions dans la procédures (dans la partie de définition de la procédure).

Détails

Nom de la procédure: MajMontantTotalCommande

Type: PROCEDURE

Direction	Nom	Type	Taille/Valeurs*	Options
IN	p_numComm:	DECIMAL	15	Supprimer
IN	p_nouveauMo	DECIMAL	15	Supprimer

Ajouter un paramètre

```

1 BEGIN
2   UPDATE Commande
3   SET montant = p_nouveauMontantTotal
4   WHERE numCommande = p_numCommande;
5 END

```

Définition

	numCommande	dateCommande	montantFrais	montant	numClient
<input type="checkbox"/> Éditer Copier Supprimer	202	2023-12-17	10	50	1

```

1 SET @numCommandeTest = 202;
2 SET @nouveauMontantTotalTest = 500.00;
3
4 CALL MajMontantTotalCommande(@numCommandeTest, @nouveauMontantTotalTest);
5
6 SELECT *
7 FROM Commande
8 WHERE numCommande = @numCommandeTest;

```

```

SET @numCommandeTest = 202;
SET @nouveauMontantTotalTest = 500.00;

```

```
CALL MajMontantTotalCommande(@numCommandeTest, @nouveauMontantTotalTest);
```

```

SELECT *
FROM Commande
WHERE numCommande = @numCommandeTest;

```

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0008 seconde(s).)

```
SET @numCommandeTest = 202;
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0009 seconde(s).)

```
SET @nouveauMontantTotalTest = 500.00;
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0098 seconde(s).)

```
CALL MajMontantTotalCommande(@numCommandeTest, @nouveauMontantTotalTest);
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0.0031 seconde(s).)

```
SELECT * FROM Commande WHERE numCommande = @numCommandeTest;
```

☐ Profilage [Éditer en ligne] [Éditer] [Expliquer SQL] [Créer le code source PHP] [Actualiser]

☐ Tout afficher | Restaurer l'ordre des colonnes | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

+ Options

	numCommande	dateCommande	montantFrais	montant	numClient	idDescription
<input type="checkbox"/> Éditer <input type="checkbox"/> Copier <input type="checkbox"/> Supprimer	202	2023-12-17	10	500	1	NULL

Nom de la procédure: AjouterAdresse

Type: PROCEDURE

	Direction	Nom	Type	Taille/Valeurs*	Options
+	IN	p_idAdresseLivraison	DECIMAL	15	<input type="checkbox"/> Supprimer
+	IN	p_adrPostaleLivraison	VARCHAR	5	Jeu <input type="checkbox"/> Supprimer
+	IN	p_adrVilleLivraison	VARCHAR	50	Jeu <input type="checkbox"/> Supprimer
+	IN	p_adrPaysLivraison	VARCHAR	50	Jeu <input type="checkbox"/> Supprimer
+	IN	p_adrLivraison	VARCHAR	50	Jeu <input type="checkbox"/> Supprimer

Ajouter un paramètre

```

1 BEGIN
2   INSERT INTO Adresselivraison(idAdresseLivraison, adrPostaleLivraison, adrVilleLivraison,
3     adrPaysLivraison, adrLivraison)
4     VALUES (p_idAdresseLivraison, p_adrPostaleLivraison, p_adrVilleLivraison, p_adrPaysLivraison,
5       p_adrLivraison);
6 END

```

Définition

```
1 SET @idAdresseLivraison = 3;  
2 SET @adrPostaleLivraison = '31200';  
3 SET @adrVilleLivraison = 'Toulouse';  
4 SET @adrPaysLivraison = 'France';  
5 SET @adrLivraison = '12 rue jean sais rien';  
6  
7 CALL AjouterAdresse(  
8     @idAdresseLivraison,  
9     @adrPostaleLivraison,  
10    @adrVilleLivraison,  
11    @adrPaysLivraison,  
12    @adrLivraison  
13 );  
14  
15 SELECT *  
16 FROM AdresseLivraison  
17 WHERE idAdresseLivraison = @idAdresseLivraison;
```

```
SET @idAdresseLivraison = 3;  
SET @adrPostaleLivraison = '31200';  
SET @adrVilleLivraison = 'Toulouse';  
SET @adrPaysLivraison = 'France';  
SET @adrLivraison = '12 rue jean sais rien';
```

```
CALL AjouterAdresse(  
    @idAdresseLivraison,  
    @adrPostaleLivraison,  
    @adrVilleLivraison,  
    @adrPaysLivraison,  
    @adrLivraison  
);
```

```
SELECT *  
FROM AdresseLivraison  
WHERE idAdresseLivraison = @idAdresseLivraison;
```

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0009 seconde(s).)

```
SET @adrLivraison = '12 rue jean sais rien';
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0341 seconde(s).)

```
CALL AjouterAdresse( @idAdresseLivraison, @adrPostaleLivraison, @adrVilleLivraison, @adrPaysLivraison, @adrLivraison );
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0.0037 seconde(s).)

```
SELECT * FROM AdresseLivraison WHERE idAdresseLivraison = @idAdresseLivraison;
```

☐ Profilage [Éditer en ligne] [Éditer] [Expliquer SQL] [Créer le code source PHP] [Actualiser]

☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

+ Options

	idAdresseLivraison	adrPostaleLivraison	adrVilleLivraison	adrPaysLivraison	adrLivraison
<input type="checkbox"/> Éditer	31200	Toulouse	France	12 rue jean sais rien	

Comme cette procédure est vraiment très simple j'ai créé une fonction en plus

Nom de la procédure: CalculerMontantTotalCommande

Type: FUNCTION

Paramètres	Nom	Type	Taille/Valeurs*	Options
\$	p_numCommande	DECIMAL	15	<input type="checkbox"/> Supprimer

Ajouter un paramètre

Type de retour: DECIMAL

Longueur/valeurs de retour: 15,0

Options de retour:

```

1 BEGIN
2   DECLARE v_montantTotal DECIMAL(15);
3
4   SELECT SUM(LC.quantiteCommandee * P.prixVente) INTO v_montantTotal
5   FROM LigneCde LC
6   JOIN Produit P ON LC.numProduit = P.numProduit
7   WHERE LC.numCommande = p_numCommande;
8
9   RETURN v_montantTotal;
10 END

```

```

1 SET @numCommande = 202;
2 SELECT CalculerMontantTotalCommande(@numCommande) AS MontantTotal;

```

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0008 seconde(s).)

```
SET @numCommande = 202;
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0.0037 seconde(s).)

```
SELECT CalculerMontantTotalCommande(@numCommande) AS MontantTotal;
```

☐ Profilage [Éditer en ligne] [Éditer] [Expliquer SQL] [Créer le code source PHP] [Actualiser]

☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

+ Options

MontantTotal
2988

le produit numProduit = 2 coûte 249 et dans la commande la quantiteCommandee est de 12
 $249 \times 12 = 2988$

à noter que je compte rajouter plusieurs autres procédures et trigger mais que je n'ai pas eu le temps de les développer ou d'en faire les tests d'où le fait qu'ils ne se trouvent pas dans ce document. J'avoue avoir eu quelques difficultés du fait qu'on avait jamais utilisé phpMyAdmin avant et que j'ai passé énormément de temps à chercher où étaient mes erreurs puisque la syntaxe n'est pas la même que sur Oracle. Merci de votre compréhension.