



## Compte Rendu Base de données (Sprint 5)

GUILHAMASSÉ Quentin

LAHITTE Mathis

MARCHANT Adriel

MAZEAU Antoine

QUINVEROS Jordan

# Table des matières

- Compte Rendu Base de données Semaine 1.....1
  - 1. Mise à jour de la base .....3
  - 2. Ajout d'une procédure .....3
  - 3. Ajout de triggers .....3
    - 1) Trigger AFTER.....3
    - 2) Trigger BEFORE.....4

## 1. Mise à jour de la base

```
ALTER TABLE Article
ADD stockA NUMBER;
```

```
UPDATE Article
SET stockA = 100;
```

```
ALTER TABLE Article
ADD nbEtoilesA NUMBER;
```

```
UPDATE Article
SET nbEtoilesA = 4;
```

```
ALTER TABLE Article
ADD CONSTRAINT ck_Article_stock CHECK (stockA >= 0);
```

```
ALTER TABLE Article
ADD CONSTRAINT ck_Article_nbEtoiles CHECK ( nbEtoilesA >= 0 AND nbEtoilesA <=
5);
```

- Modification de la clé primaire de la table Domicile ( adresseD → idD)
- Création de la table Panier

```
DROP TABLE Panier;
CREATE TABLE Panier
(
    numC NUMBER,
    refA VARCHAR(30),
    quantite NUMBER,
    CONSTRAINT pk_Panier PRIMARY KEY (numC,refA),
    CONSTRAINT fk_Panier_numC FOREIGN KEY (numC) REFERENCES Client(numC),
    CONSTRAINT fk_Panier_refA FOREIGN KEY (refA) REFERENCES Article(refA),
    CONSTRAINT ck_Panier_quantite CHECK (quantite >0)
);
```

## 2. Ajout d'une procédure

```
-- procédure qui remet du stock sur un article sélectionné
CREATE OR REPLACE PROCEDURE remettre_du_stock ( param1 Article.refA%TYPE) AS
BEGIN
    UPDATE Article
    SET stockA = 100
    WHERE refA = param1;
END;
```

## 3. Ajout de triggers

### 1) Trigger AFTER

```
-- Trigger qui met à jour le stock d'un article quand un insert sur Constituer
est effectué
-- lève une erreur quand la quantité à déduire passe met le stock en négatif
CREATE OR REPLACE TRIGGER t_u_Article_majStock
AFTER INSERT ON Constituer
```

```

FOR EACH ROW
DECLARE
    ERR_STOCK_INSUFFISANT EXCEPTION;
    PRAGMA EXCEPTION_INIT( ERR_STOCK_INSUFFISANT, -20001 );
    v_stockA Article.stockA%TYPE;
BEGIN
    SELECT stockA INTO v_stockA
    FROM Article
    WHERE refA = :NEW.refA;

    IF (v_stockA - :NEW.quantite < 0 ) THEN
        RAISE ERR_STOCK_INSUFFISANT;
    END IF ;

    UPDATE Article
    SET stockA = stockA - :NEW.quantite
    WHERE refA = :NEW.refA;
EXCEPTION
    WHEN ERR_STOCK_INSUFFISANT THEN
        DBMS_OUTPUT.PUT_LINE('Le stock de 1 article est insuffisant pour
cette commande');
END;

```

## 2) Trigger BEFORE

```

-- Trigger BEFORE qui avant une commande vérifie si il y a assez de stock et
remet 100
CREATE OR REPLACE TRIGGER t_u_Article_stockVide
BEFORE INSERT ON Constituer
FOR EACH ROW
DECLARE
    v_stockA Article.stockA%TYPE;
    v_refA Article.refA%TYPE;
BEGIN

    SELECT stockA INTO v_stockA
    FROM Article
    WHERE refA = :NEW.refA;

    SELECT refA INTO v_refA
    FROM Article
    WHERE refA = :NEW.refA;
    IF (v_stockA <= 0) THEN

        remettre_du_stock(:NEW.refA);
    END IF;
END;

```