

Groupe G1A-1

S3.A.01 - Développement appli

Livrable BD semaine 2

Table des matières

1 Introduction	3
2 Livrable	3
a. Code de création	3
b. Résultat	4

S3.A.01 - Développement appli

Livrable BD semaine 2

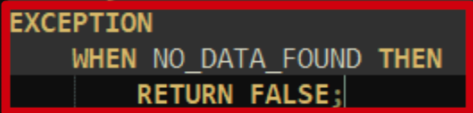
1 Introduction

Ce livrable a pour but de montrer la gestion des exceptions dans les deux procédures PL/SQL stockées.

2 Livrable

a. Code de création

```
CREATE OR REPLACE FUNCTION hasOrdered (  
  p_client Clients.idClient%TYPE,  
  p_produit Produits.refProduit%TYPE  
)  
  RETURN BOOLEAN  
IS  
  v_quantite Constituer.quantite%TYPE;  
BEGIN  
  SELECT DISTINCT quantite into v_quantite  
  FROM Constituer con, Commandes com, Produits pro  
  WHERE con.refcommande = com.refcommande  
  AND con.refproduit = p_produit  
  AND com.idclient = p_client;  
  
  IF (v_quantite IS NOT NULL) THEN  
    RETURN TRUE;  
  ELSE  
    RETURN FALSE;  
  END IF;  
  
  EXCEPTION  
    WHEN NO_DATA_FOUND THEN  
      RETURN FALSE;  
END hasOrdered;  
/
```




Une première fonction ayant pour but de déterminer si un client à déjà commandé un produit. Ce qui nous permettra de vérifier lorsque celui-ci voudra déposer un avis sur le produit en question. La gestion de l'exception a été gérée grâce au code entouré en rouge.

```

CREATE OR REPLACE PROCEDURE addToOrder (
  p_order Commandes.refcommande%TYPE,
  p_prod Produits.refproduit%TYPE,
  p_quantite Constituer.quantite%TYPE
)
IS
  fkErreur EXCEPTION;
  PRAGMA EXCEPTION_INIT(fkErreur, -2291);
  ckErreur EXCEPTION;
  PRAGMA EXCEPTION_INIT(ckErreur, -2290);
BEGIN
  INSERT INTO Constituer
  VALUES (p_order, p_prod, p_quantite);
  COMMIT;
EXCEPTION
  WHEN ckErreur THEN
    DBMS_OUTPUT.PUT_LINE('erreur de contrainte check');
  WHEN fkErreur THEN
    DBMS_OUTPUT.PUT_LINE('erreur de clé étrangère');
  WHEN DUP_VAL_ON_INDEX THEN
    DBMS_OUTPUT.PUT_LINE('erreur de clé primaire');
END;
/

```



Et une deuxième procédure qui a pour but d'ajouter un produit avec une certaine quantité dans une commande donnée. La gestion des exceptions a été correctement gérée grâce au code entouré en rouge.

b. Résultat

Elément Function HASORDERED compilé

Elément Procedure ADDTOORDER compilé

Les deux compilent correctement.