

Semestre : 1 ☐ 2 ☒

Session : Principale ☒ Rattrapage ☐

Module: Système de Gestion des Bases de Données

Documents autorisés : OUI ☐ NON ☒

Enseignants: H. Bellakhdhar, H. Sfaxi, I. Baklouti, I. Channoufi, I. Slimene, J. Hkimi, M. Jebari, S. Refai, S. Yekken

Nombre de pages: 2

Date: 18/05/2015

Heure: 09h00

Durée: 1h30

Classes: 3A, 3B

Barème: 4-5-4-1-3-3

Soit le schéma relationnel suivant :

Client (cin, nom, adresse, salaire)

Banque (code_banque, nom_banque)

Agence (#code_banque, code_agence, adresse)

Compte (RIB, #(code_banque, code_agence), date_ouverture, #cin)

Dossier_Credit (code_dossier, #RIB, date_acceptation, montant_credit)

1. a. Ecrire une fonction **FN_ANC (p_cin in number)** qui permet de calculer le nombre d'années d'ancienneté d'un client entré en paramètre, sachant qu'un client ne peut avoir qu'un seul compte.
b. Traduire cette fonction en une procédure stockée **PROC_ANC**.
2. Ecrire une procédure **PS_CREDIT(p_cin in number, val_credit in number, nbre_annees in number)** qui permet de calculer la valeur d'échéance d'un crédit. Pour ce faire, la procédure doit vérifier les conditions d'acceptation d'un dossier de crédit et gérer les exceptions suivantes :
 - i. Vérifier l'existence du client.
 - ii. Ancienneté du client supérieure à 1 année (utiliser la fonction **FN_ANC**).
 - iii. Salaire du client supérieur à 1500.

Si les conditions sont satisfaites, la procédure doit calculer et afficher la valeur d'échéance selon la formule suivante :

$$\text{val_echeance} = \text{val_credit} / (\text{nbre_annees} * 12)$$

3. Sachant qu'un client peut avoir plus qu'un dossier de crédit, écrire une procédure stockée **PS_DOSSIERS** (**p_code_banque in number**) qui permet d'afficher pour une banque donnée en paramètre, la liste de ses clients ainsi que la liste de leurs dossiers.
4. Ecrire un bloc anonyme qui permet d'afficher les clients de la banque dont le code est '183' ainsi que leurs dossiers.
5. Ecrire une procédure **PS_NBRE_DOSSIERS** qui permet d'afficher pour chaque banque le nombre de dossiers de crédit acceptés durant le 1^{er} semestre de l'année courante.
6. Ecrire un trigger **TRIG_RIB** qui, avant insertion dans la table **COMPTE**, calcule le **RIB** en se basant sur la formule suivante :

$RIB = code_banque || code_agence || cin || cle_rib$

$Cle_rib = 97 - (MOD((89 * code_banque + 15 * code_agence + 3 * cin), 97))$

.....ANNEXE.....

```
CREATE[OR REPLACE] FUNCTION
nom_fonction [(argument1,...argumentN)]
RETURN type_donnée
IS
[Déclaration des variables locales]
BEGIN
<Instructions;>
Return valeur;
END;

CREATE[OR REPLACE] PROCEDURE
nom_procedure [(argument1,...argumentN)]
IS
[Déclaration des variables locales]
BEGIN
<Instructions;>
END;
```

```
CREATE[OR REPLACE] TRIGGER nom_trigger
{BEFORE | AFTER | INSTEADOF}
{INSERT | UPDATE {of COLUMN} | DELETE}
OR {INSERT | UPDATE {of COLUMN} | DELETE}
ON nom_table
FOR EACH ROW
WHEN
condition
DECLARE
<declarations>
BEGIN
<Instructions ;>
END;
```

Les exceptions prédéfinies :

Nom Exception	Erreur Oracle
NO_DATA_FOUND	-01403
TOO_MANY_ROWS	-01422
ZERO_DIVIDE	-01476
DUP_VAL_ON_INDEX	-00001