

EPSI Grenoble B3 DNT 2020/21 - Projet Oracle / PL-SQL

Consignes

Travail à réaliser en groupe de maximum 4 personnes. Envoyer le script de création des tables et du package, ainsi que les jeux d'essais, à l'adresse mathieu.mulot2@reseau-cd.net, avec pour objet « EG3 PL/SQL ». Rappeler les membres du groupe dans le nom du script et le corps de l'e-mail.

Les critères d'évaluation seront :

- Pas d'erreurs d'exécution des scripts
- Résultats corrects
- Qualité du code (nommage des variables, algorithmes, ...)
- Qualité des jeux d'essais

Enoncé

Dans Oracle Database, il est possible d'envoyer des e-mails et de planifier des travaux.

Le but du projet est de mettre en place une base de données Oracle et les éléments PL/SQL permettant de planifier l'envoi d'e-mails et de réaliser des tentatives ultérieures en cas d'échecs. NB : nous n'implémentons pas ici l'envoi et la planification réels des e-mails, mais nous mettons en place les éléments pour que cela puisse être réalisé en utilisant les fonctionnalités intégrées à Oracle (cf. liens dans fichier txt joint au sujet).

Pour cela, les tables avec toutes les contraintes nécessaires, ainsi qu'un package PL/SQL, doivent être créés avec les éléments suivants :

- Une exception EMAIL_EXCEPTION qui servira aux identifiants d'e-mails inexistant indiqués lors des tentatives d'envois (procédure sendMails)
- Un RECORD nommé RMAIL représentant un e-mail à envoyer avec un sujet, un texte, une liste de destinataires principaux, une liste de destinataires en copie, une liste de destinataires en copie cachée et la date d'envoi souhaitée avec heure
- Une procédure createMails prenant en paramètre un tableau d'éléments de type RMAIL et qui insère les enregistrements correspondant dans la table des travaux TMAIL, qui comporte les mêmes colonnes que les champs du RECORD RMAIL avec en plus un identifiant
- Une procédure checkMail prenant en paramètre un identifiant d'e-mail et qui lève l'exception EMAIL_EXCEPTION s'il ne correspond à aucun e-mail existant
- Une procédure sendMails qui récupèrent les identifiants de tous les e-mails en échec, ainsi que ceux qui n'ont pas encore été envoyés et dont la date de planification est passée, et ceux passés en paramètre (tableau d'identifiants) et qui correspondent à des e-mails prioritaires (date d'envoi souhaitée non encore dépassée ou sans aucune tentative d'envoi infructueuse). Cette procédure serait planifiée à intervalles réguliers faibles (toutes les minutes par exemple). Elle utilise la procédure checkMail pour logger toutes les tentatives d'envoi sur identifiant d'e-mail inexistant
- Une procédure cleanMails prenant en paramètre une date et qui supprime tous les e-mails dont la date d'envoi souhaitée est dépassée

- Une table TMAILS_SEND_HISTO contenant un historique d'envoi des e-mails, avec une clé primaire, l'identifiant de l'e-mail, la date de la tentative d'envoi et le statut (réussite ou échec). La suppression d'un e-mail provoquera automatiquement la suppression de son historique d'envois (par trigger, pas de CASCADE)
- Une table TMAILS_LOGS, alimentée par trigger(s), et contenant les logs de toutes les opérations sur les e-mails réalisées (ajout/modification/suppression/tentatives d'envoi), avec un message (de réussite ou d'erreur) précisant les identifiants des e-mails et la date de l'opération. Elle contient une clé primaire, un message et la date du log