

# TP JDBC

## REQUETES STATIQUES ET PARAMETREES EXECUTION DE RESSOURCES PL /SQL

Répondre sous forme d'application JAVA aux questions suivantes :  
(et accéder via JDBC à la base de données)

**JD.01 )** Afficher la liste des plantes (identifiant et couleur) ?

**JD.02 )** Afficher la liste des plantes dont la date de floraison est supérieure à une certaine date.

---

## UNE ARCHITECTURE LOGICIELLE EN COUCHE

### EXPLOITATION D'UN NOYAU APPLICATIF BASÉ SUR DES PROCÉDURES STOCKÉES

Les architectures applicatives font souvent intervenir plusieurs couches logicielles. On va notamment trouver :

- la couche **IHM**
- la couche **Traitement** (+ accès ressources BD : API comme JDBC)
- la couche **Noyau applicatif** (càd un ensemble de ressources déployées côté serveur proche des données et des règles de gestion, évitant des allers-retours intempestifs et lourds sur le réseau)
- la couche **BD**

On exploite JDBC, couche d'abstraction d'accès aux données pour Java : communiquer depuis une application JAVA avec une BD et en exploiter des ressources telles les procédures stockées.

Une telle architecture permet de répartir les développements selon leur nature et de profiter des avantages des procédures stockées dans une application distribuée.

⇒ **VOUS DEVEZ REALISER :** *par équipe de 2 personnes*

### **JD.03 ) UNE APPLICATION JAVA DE GESTION DE COMPTES BANCAIRES**

L'**application Java** doit consister en un menu aiguillant vers les différentes possibilités offertes par le **noyau applicatif GESTCOMPTES**, à savoir :

- la création de compte,
- le dépôt sur un compte,
- le retrait depuis un compte,
- le virement depuis un compte vers un autre compte,
- le solde d'un compte,
- la clôture d'un compte.

Les **retours en erreur** de ces procédures et fonctions-là seront traités, c'est-à-dire **capturés via JDBC**. Par exemple quand le compte n'existe pas.

Vous devrez offrir aussi la possibilité de **connaître l'historique d'un compte**, et ceci selon deux modes (dernières opérations ou derniers jours) avec comme argument le nombre d'unité dans le mode. Par exemple, `historique('FRM01','dernoper',6)` renvoie les 6 dernières opérations pour ce compte-là.

#### **=> AU PRÉALABLE, INSTALLEZ LES RESSOURCES BD :**

##### **1) Récupérer sous SUPPORT\BD\JDBC\**

- le script *creeBDgestcompte.sql*

Création du schéma relationnel.

Il contient les données sur les comptes et assure le stockage des données de l'activité.

- le script *creePCKgestcomptespec.sql*

Spécification du paquetage "gestcomptes".

- le script *creePCKgestcomptebody.sql*

Corps du paquetage "gestcomptes".

**GESTCOMPTES** est codé en PL/SQL

Sous la forme d'un paquetage contenant des procédures stockées et des fonctions.

##### **2) Sous SQL, exécuter ces scripts**

```
start f:\creeBDgestcompte.sql;
```

```
start f:\creePCKgestcomptespec.sql;
```

```
start f:\creePCKgestcomptebody.sql;
```