

Projet Viticulture TP 1 : découverte

Partie 1 : bases de données locales SQLite

Partie 2 : projet H2O – stockage local

Partie 3 : bases de données distantes

Partie 4 : projet H2O – synchronisation avec un serveur distant

Description du thème

Ce TP permet d'aborder par la pratique les techniques permettant de stocker des données dans des bases SQLite sous Android.

Propriétés	Description
Intitulé long	Projet Viticulture - TP 1 : découverte Comprendre le fonctionnement des bases de données SQLite
Formation concernée	BTS Services informatiques aux organisations
Matière	SLAM 4, PPE
Présentation	Les TP permettent de découvrir différentes méthodes pour accéder et utiliser des bases de données SQLite dans un environnement mobile Android
Notions	Savoirs <ul style="list-style-type: none">• D4.1 - Conception et réalisation d'une solution applicative• D4.2 - Maintenance d'une solution applicative Savoir-faire <ul style="list-style-type: none">• Programmer un composant logiciel• Exploiter une bibliothèque de composants• Adapter un composant logiciel• Valider et documenter un composant logiciel• Programmer au sein d'un framework
Pré-requis	Développement mobile sous Android
Outils	Eclipse, framework Android API 18
Mots-clés	Application mobile, Android, SQLite
Durée	4h
Auteur(es)	Mathieu Capliez
Version	v 1.0
Date de publication	Octobre 2014

Énoncé

Documentation, les javadocs de l'interface Cursor et de la classe SQLiteDatabase :

<http://developer.android.com/reference/android/database/Cursor.html>

<http://developer.android.com/reference/android/database/sqlite/SQLiteDatabase.html>

Dans le cadre d'un concours, les producteurs de vin de la région Aquitaine sont évalués selon la qualité gustative de leur production.

Lors de la dégustation, chaque membre du jury attribue une note au viticulteur pouvant aller de 0 à 100. Si le jury souhaite modifier cette dernière, ou s'il goûte un autre vin du viticulteur, il pourra changer la note associée.

Les applications Android peuvent faire appel à des bases de données SQLite pour stocker de manière locale des données. Chaque application peut accéder à une ou plusieurs bases, de même Android permet d'en gérer plusieurs versions.

Dans cet exercice de découverte, différentes méthodes d'accès aux bases de données SQLite seront étudiées. Vous pourrez ainsi déterminer les pratiques les plus adaptées à vos projets.

Après avoir importé les deux projets "bd1" et "bd2" fournis, tester et observer les codes afin de répondre aux questions suivantes :

Partie 1 : Base de données

Question 1 – Analyse de la classe `BdSQLiteOpenHelper`

A l'aide du projet `bd1` :

- 1.1. Quel est le rôle des instructions lignes 25, 27, et 29 à 32 ?
- 1.2. Identifier le rôle de la méthode `onCreate ()` et indiquer à quel moment cette méthode est exécutée.
- 1.3. A l'aide de la documentation Android expliquer le rôle de la méthode `onUpgrade ()` .
- 1.4. Proposer le schéma relationnel de la base de données utilisée dans cette application.
- 1.5. Quel est le rôle de la classe `BdSQLiteOpenHelper` héritant de `SQLiteOpenHelper` ? Quels sont les rôles des paramètres de son constructeur ?

Question 2 – Accès aux données

- 2.1. Quel est le rôle de la méthode `methode1 ()` dans la classe `MainActivity`?
- 2.2. Quel type d'opération sur la base de données est effectué lors des appels `bd.rawQuery ()` et lors des appels `db.execSQL ()` ?
- 2.3. Modifier le code de la méthode `methode1 ()` du projet `BD1` afin d'afficher l'ensemble des données - `idV`, `nomV`, `niveauV` - de la table `viticulteur` dans une liste `scrollable` (`ScrollView`).
- 2.4. Comparatif avec le projet `BD2` : indiquer pour chaque projet dans quelle classe se situe l'accès aux données.

Partie 2 : Application

Question 3 – Analyse des méthodes `addViticulteur1 ()` et `addViticulteur2 ()` de la classe `ViticulteurDAO`

- 3.1. Quel est le rôle de chacune de ces 2 méthodes ?
- 3.2. Quelles sont les différences dans l'implémentation de cette fonctionnalité ?
- 3.3. D'où provient la valeur retournée par la méthode `addViticulteur2 ()` ? Que contient la variable dont la valeur est retournée ?
- 3.4. Dans quelle situation peut servir cette valeur ?

3.5. En utilisant la documentation Android, indiquer quelle est la valeur retournée par la méthode `insert()` de la classe `SQLiteDatabase`.

3.6. Pourquoi l'accès à la base peut-il être fait en lecture seule ou en lecture-écriture ? Quelles méthodes donnent ces accès ?

Question 4 – Adaptation de la solution

Dans les 2 projets :

4.1. Dans le cadre du concours et pour faciliter le travail des jurys, mettre en place la fonctionnalité permettant de modifier le niveau d'un viticulteur selon son identifiant. Décrire quelle requête SQL doit être exécutée, et dans quelle classe l'intégrer.

4.2. Les membres du jury souhaitent pouvoir connaître la position d'un viticulteur dans le classement. Mettre en place une méthode permettant de récupérer une liste de viticulteurs dont le niveau est meilleur que celui passé en paramètre.

Partie 3 : Réflexion

Question 5 – Analyse de la structure applicative

5.1. Quel est le rôle de la classe `ViticulteurDAO` ?

5.2. Est-il possible d'utiliser la classe `ViticulteurDAO` dans un autre projet ?

5.3. Quels avantages et inconvénients y a-t-il à créer une classe DAO ?

Question 6 – Réflexion sur la gestion des accès aux bases de données

6.1. On souhaite faire appel à un SGBDR différent. Dans ces deux projets, indiquer quelles méthode et classes sont à modifier.

6.2. La société souhaite utiliser un code similaire pour accéder à une base de données sur une nouvelle application Android. Quelle méthode d'accès est la moins liée au code applicatif ?