

Challenge Cooking Recipes

Utiliser une base de données MySQL et développer un requêteur en ligne de commande avec une mise en forme textuelle.

Une base de donnée de recette composé de 3 tables vous ai fourni :

- recipes - Contient les recettes
- quantity - Contient les quantités des ingrédients lié aux recettes
- ingredients - Contient les ingrédients avec leurs prix

[Schema de la base de données a importé.](#)

Votre programme devra permettre :

- D'exécuter des requêtes MySQL pour rechercher des recettes
- Afficher sous forme de tableau en console les résultats de la recherche (mode search)
- Affichage d'une recette simple via son ID (mode recipe)
- Implémenter des filtres et tris sur la recherche
- Chercher des recettes par ingrédient disponible (mode leftovers)
- Limiter les résultats de recherche à un prix de revient maximum

Proposition d'architecture

Votre programme peut-être décomposé en 3 parties/modules:

- Analyse des arguments du programme
 - Solution 1 Utiliser les arguments du programme argv
 - Solution 2 Utiliser la fonction scanf et construire des menus interactifs (plus long)
- Requête SQL en fonction du mode sélectionné
- Affichage des résultats

Les outils à utiliser

Xampp ou Wamp ou Mamp ou Lamp

Cela vous permettra d'avoir un serveur MySQL prêt à l'emploi ainsi que l'interface d'administration PhpMyAdmin.

<https://www.apachefriends.org/index.html>

<https://alternativeto.net/software/xampp/>

MySQL

Outils de gestion de base de donnée.

Documentation de l'API C :

<https://dev.mysql.com/doc/refman/5.6/en/c-api.html>

https://github.com/hholzgra/connector-c-examples/blob/master/mysql_fetch_row.c

<https://github.com/hholzgra/connector-c-examples/>

Démarrage du projet sous Windows (CodeBlocks)

1. Installer CodeBlocks avec MinGW télécharger et installer codeblocks-16.01mingw-setup.exe <http://www.codeblocks.org/downloads/binaries>
2. Utiliser le [template du projet](#) pour CodeBlocks avec les librairies déjà installées et configurées.

Démarrage du projet sous Mac OSX (Xcode)

1. Ouvrir un terminal
2. Installer [Homebrew](#) ``/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"``
3. Installer mysql-connector-c ``brew install mysql-connector-c``
4. Démarrer Xcode
5. Créer un Projet Xcode C
6. Configurer les Build Settings `Search Paths > Header Search Paths` ``/usr/local/opt/mysql-client/include``
7. Configurer les Build Settings `Search Paths > Library Search Paths` ``/usr/local/opt/mysql-client/lib``
8. Configurer les Build Settings `Linking > Other Linker Flags` ``-lmysqlclient``

Démarrage du projet sous Ubuntu

1. Ouvrir un terminal
2. Installer make, libcurl, libjson-c ``apt install make libmysqlclient-dev``
3. Télécharger le [Makefile](#) d'exemple
4. Lancer la commande ``make``

Barème de notation

- Génie logiciel 10 points
 - Respect des conventions de codage
 - Qualité du code (Requêtes préparés avec les fonctions stmt)
 - Qualité de l'affichage
 - Travail en groupe
- Fonctionnalités 10 points
 - Lancer des requêtes SQL
 - Afficher sous forme de tableau les résultats de la recherche
 - Implémenter des filtres sur la recherche
 - Voir le détails d'un élément sous forme de texte sur demande
 - Implémenter des tries sur les colonnes via une option sort
 - Implémenter le mode search
 - Implémenter le mode recipe
 - Implémenter le mode leftovers