**ML-DK-APP**

A simple data management / search web application with MarkLogic Java API

**概要**

MarkLogic 7を使ったサンプルプログラムです。

[MarkLogic](http://www.marklogic.com/) はMongoDBに全文検索とトランザクションがくっついたようなドキュメントストア型のNoSQLデータベースです。  
使い勝手が良くて枯れているんですが日本語の技術情報が少ないようなので、やりたいことを探す逆引き的なリファレンスとしてコードを公開してみます。

Java APIやRESTを使ってデータベースクライアント側でロジックを書いてもいいですし、XQueryを使ってデータベース側でアプリケーションを作ってもよいですが、データのCRUDや検索だけであればお手軽な前者を使えば十分です。  
このサンプルプログラムではMarkLogicのJava APIを使って、Javaで簡単なWebアプリケーションを作っています。

Java APIのリファレンスは[こちら](https://docs.marklogic.com/javadoc/client/index.html)で、ライブラリの取得は[こちら](https://developer.marklogic.com/products/java)です。

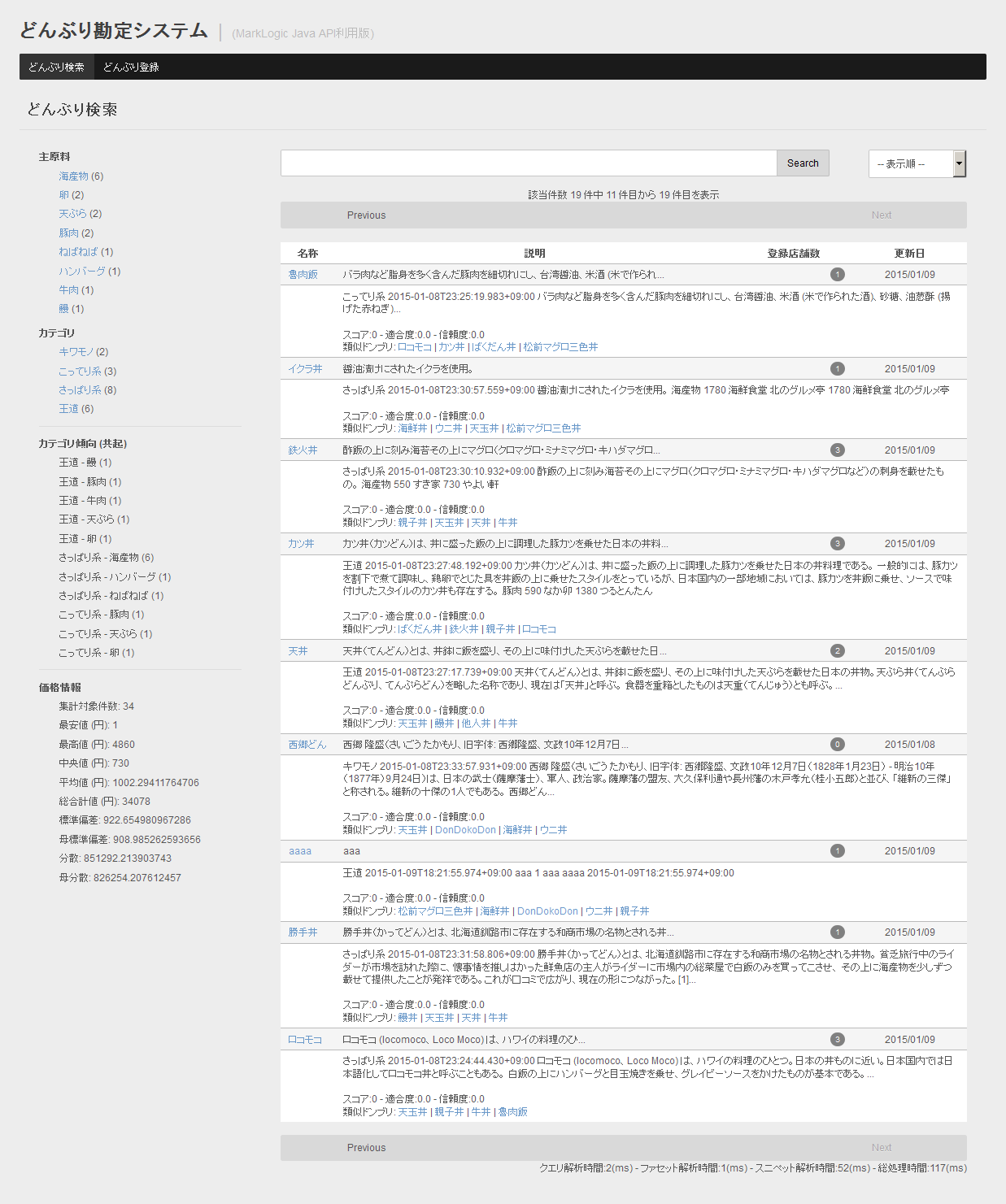
あくまでサンプルですので、決まり文句ではありますが、ご利用にあたって発生した不利益については免責とさせていただきますのであしからずご了承ください。

**説明**

以下のコードを含んでいます。

* データCRUD
* トランザクション (commit/rollback)
* フルテキスト検索
* 共起抽出
* MapReduce集計

**スクリーンショット**

メイン画面 (ファセットと共起と集計値を配置しています。)  
[](https://github.com/ytsejam5/ML-DK-APP/blob/master/screenshot/dk-app-screenshot-001.png)

登録・更新画面 (1対Nで親子関係を持っててもそのままストアできる、という例です。)  
[](https://github.com/ytsejam5/ML-DK-APP/blob/master/screenshot/dk-app-screenshot-002.png)

**処理の流れ**

メイン画面での表示項目は[ListAction.java](https://github.com/ytsejam5/ML-DK-APP/blob/master/dk-app/WEB-INF/src/com/github/ytsejam5/dk/action/ListAction.java)でMarkLogicから取得しています。  
検索条件もろもろを[query-options.xml](https://github.com/ytsejam5/ML-DK-APP/tree/master/dk-app/WEB-INF/src)で指定していまして、ファセット、共起、集計値、検索結果(1件のuri)、スニペットあたりを1リクエスト-1レスポンスで取得しています。  
検索条件のオプションのリファレンスは[こちら](https://docs.marklogic.com/guide/rest-dev/appendixb)ですね。

POJOの出し入れは[action](https://github.com/ytsejam5/ML-DK-APP/tree/master/dk-app/WEB-INF/src/com/github/ytsejam5/dk/action)下です。  
write(pojo) / pojo = read() 的な感じです。POJOをそのままストアできるのでOR Mapperは要らないですね。

**インストール**

1. Application Servicesでデータベース作成
2. 以下のElement Range Indexを設定
   * created,updated (namespaceなし / dateTime型)
   * price (namespaceなし / dateTime型)
   * shop,name,category,ingredient (namespaceなし / string型)
3. Application ServicesでRESTインタフェース作成
4. [MLCP](http://developer.marklogic.com/products/mlcp)でデータロード (接続先と-input\_file\_pathオプションくらいでOKです。)
5. Java APIのライブラリを取得。[こちら](https://developer.marklogic.com/products/java)からです。
6. dk-app/WEB-INF/lib下に放り込みます。
7. お好みのアプリケーションサーバにdk-app以下をデプロイ (Tomcat 8.0.8で動作確認しています。)
8. /dkにアクセス！