**Scava Test Assignment 1:**

a. Downloading a 750MB CSV file everyday from an FTP site

The file contains 50,000 rows. Each row has the following information

 \_Product id

 \_Product name

 \_Product description

 \_Average rating

 \_Top 3 reviews

 \_Regular Price

 \_Sale Price

 \_Colors and Sizes offered

 \_Primary Image

 \_Alternate Images

**Assumptions:**

1. Data is unique, no duplicates checking.
2. Clearing old data, no attempt to check if product id exists and selectively inserting records.
3. No authentication.
4. Rollback, timeout, pagination, error handling is handled by the server end.
5. Primary image and Alternate image are links.

**Pre steps:**

1. Create local database on local application end to hold data.
2. Clear the database when first insertion is initiated.

**OR**

1. Use no-SQL database on local application end to hold data.
2. Clear the data in this structure before first insertion.
3. Can use Asynchronous threads to create records. (performance +)

**One Possible Way:** Application initiates FTP pull process during off peak time.

**commonSteps:**

FTP the entire file on application local application file directory. Restart if error occurred.

Open the local file in file directory.

Implement line-by-line parser (since it is CSV and not xml) till End of file.

Parse the file. Create record like a C-struct.

Clear the database.

Insert records in database.

Delete the local file.

Handle Errors.

**Second possible Way: S**erver initiates ftp push during off peak time.

Cron job or scheduler, initiating the file transfer. Maybe a remote notification. The application then connects and pulls the file.

Follow commonSteps.

**Scalability**:Save data on multiple servers, so retrieval of data can be asynchronously retrieved from any server.

**Scava Test Assignment2:**

Exposing a web service that will return product information given a phrase. The phrase needs to be checked against product id, name and description

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Tag | Required | Description |
| scava.productInfo.search |  |  | scava.productInfo.search returns a list of products on SCAVA by id (using **product\_Id**), by name (using product\_name), by keywords (using **tags**), or a combination of the two (using tag\_mode). By setting the **text** field in your keyword, queries also search the product descriptions, in addition to searching through the products title and subtitles. This is a general-purpose web service, open to public. No authentication is needed to retrieve product from SCAVA sources. |
|  |  |  |  |
| Authentication |  |  | This method does not require authentication. |
|  |  |  |  |
| Arguments | api\_key | Yes | Your application Key. |
|  | product\_id | No | Product id. |
|  | product\_name | No | Product name. |
|  | tags | No | A comma delimited list of tags. Products with one or more tags listed will be returned. You can exclude results that match a term by prepending it with a - character. |
|  | tag\_mode | No | Either ‘any’ for an OR combination of tags or ‘all for an AND combination. Defaults to ‘any’ if not specified. |
|  | text | No | A free text search. Product whose id, name or tags contain the text will be returned. You can exclude results that match a term by prepending it with a - character. |
|  | min\_date | No | Products with date more than min\_date will be returned. |
|  | max\_date | No | Products with date less than max\_date will be returned. |
|  | sort | No | Specify the order in which returned items are sorted, such as by price or by listing max\_date. Default is max\_date descending. |
|  | min\_price | No | Products with price range more than minimum price. Default is 0. |
|  | max\_price | No | Products with price range less than maximum price. Default is ‘ANY’ |
|  | per\_page | No | Number of products returned per page. Default is 100. |
|  | page | No | The page of results to be returned. Defaults to 1. |
| Example Request:GET |  |  | http://svcs.scava.com/services/search/scava.productInfo.search/?     api\_key=YourApiKey&  text=shirt,black,men&     per\_page=2&     min\_price=20 |
| Exampl Request:POST |  |  | <?xml version="1.0" encoding="UTF-8"?>  < scava.productInfo.search xmlns="http://www.scava.com/services/search/">  <api\_key>tolkien</api\_key>  <text> shirt,black,men</text>  <per\_page>2</per\_page>  <min\_price>20</min\_price>  </scava.productInfo.search > |
| Example Response |  |  | <?xml version="1.0" encoding="utf-8"?>  < scava.productInfo.searchResponse xmlns:ms="http://www.scava.com/services/search"  xmlns="http://www.scava.com/services/search/ ">  <ack>Success</ack>  <version>1.0.0</version>  <timestamp>2009-09-10T18:22:25.286Z</timestamp>  <searchResult count="2">  <item>  <itemId>290347681206</itemId>  <title>Macy’s Shirt</title>  <globalId>MACY-US</globalId>  <subtitle>pin striped</subtitle>  <galleryURL>  http://www.scava.com/pict/2903476812068080\_1.jpg</galleryURL>  <viewItemURL>  http://www.scava.com/pict/2903476812068080\_1.jpg  </viewItemURL>  <productId type="ReferenceID">63875752</productId>  <sellingStatus>  <currentPrice currencyId="USD">21.45</currentPrice>  </sellingStatus>  </item>  <item>  <itemId>290347681206</itemId>  <title>Macy’s Shirt</title>  <globalId>MACY-US</globalId>  <subtitle>plain striped</subtitle>  <galleryURL>  http://www.scava.com/pict/2903476812068080\_1.jpg</galleryURL>  <viewItemURL>  http://www.scava.com/pict/2903476812068080\_1.jpg  </viewItemURL>  <productId type="ReferenceID">63875752</productId>  <sellingStatus>  <currentPrice currencyId="USD">21.45</currentPrice>  </sellingStatus>  </item>  </searchResult>  <paginationOutput>  <totalPages>1011</totalPages>  <totalEntries>2022</totalEntries>  <pageNumber>1</pageNumber>  <entriesPerPage>2</entriesPerPage>  </paginationOutput>  </findItemsAdvancedResponse> |
| Error Codes |  |  |  |
|  | Invalid api\_key |  | An Api\_key passed is invalid or expired. |
|  | Sorry, SCAVA search API is not available |  | The SCAVA API search databases are temporarily unavailable. |
|  | Service currently unavailable |  | The requested service is temporarily unavailable. |
|  | Invalid REST envelope |  | The REST envelope sent in request could not be parsed. |
|  |  |  |  |
| Change Updates |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |