

Programming Assignment 3

Introduction:

Have developed a web application which enables user to search the content of the programmable web service data stored in the MongoDB no sql database, there are two types of services record which have been stored in the database namely api and mashup user can select the type and based on the type he will be shown different criteria to search for e.g. Updated year, tags, protocols etc. and the results will be search based on the criteria and the text user has entered in the text box.

The application has been developed in Mvc framework using jsp and servlet, and MongoDB No SQL data in the backend. Further the application has been deployed in Apache Tomcat Server 8

Data Preprocessing:

1. Have create 2 collections in mongodb database(test) named **apis** and **mashup** which stores following records of apis/mashup in the given data format

As we have been provided two text files where each line represent the api/mashup record and various fields are separated by \$#\$, \$\$\$, ### there was need to parse each field so as to make each field separate from each other and store it in the database to ease the process of performing various operations on data, so following steps were performed

1. Split each field by \$#\$
2. Check whether that field further contains ### if yes then further split that field by ###
Store it in a list and then store that list as an JSON object
Eg. Consider, *tag1###tag2###tag3* so in this case tag1, tag2, tag3 are separated by ### so further store them in an array/list and then store. Resultant field become
Tags: [tag1,tag2,tag3]
3. If the string doesn't contain ### then directly store the string split at the index in appropriate data type Eg. int, string, float {name:"ame of the api"}
4. If the split string is empty store a null object
5. Then further check whether field contains \$\$\$ if yes then again split that field on \$\$\$ and store in the list as mentioned above
Eg. Consider,

Flickr\$\$\$http://www.programmableweb.com/api/flickr###GoogleMaps\$\$\$http://www.programmableweb.com/api/google-maps

6. In the following case data is stored as follows,

Apis:[

```
{ api:
  {
    apiName:"Flickr",
```

```

        apiUrl: www.programmableweb.com/api/flickr
    }
    Api:
    {
        apiName:" Google Maps"
        apiUrl:" http://www.programmableweb.com/api/google-maps"
    }
}
]

```

Additional Functionality:

1. The web application designed takes care of sensitivity
2. For the keyword search created an index on fields title, description and summary so as to search all the key words in at least one of these fields

Design Of the Data Structure For Apis (Note:If N/A that means field doesn't exist in the type)

| Type-----> | Api | Mashup |
|-----------------|-------------------------|-------------------------|
| Name of Field | Data Type | Data Type |
| id | String | String |
| title | String | String |
| summary | String | String |
| rating | Float | Float |
| name | String | String |
| label | String | String |
| author | String | String |
| description | String | String |
| type | int | int |
| downloads | String | String |
| useCounts | int | int |
| sampleUrl | String | String |
| downloadUrl | String | N/A |
| dateModified | Date | Date |
| numComments | int | int |
| remoteFeed | String | N/A |
| commentsUrl | String | String |
| Tags | Contains list of string | Contains list of string |
| category | String | N/A |
| protocols | String | N/A |
| serviceEndpoint | String | N/A |
| version | String | N/A |
| wsdl | String | N/A |
| Data Formats | String | N/A |

| | | |
|------------------|----------------|---|
| apigroups | String | N/A |
| example | List of String | N/A |
| clientInstall | String | N/A |
| authentication | String | N/A |
| ssl | String | N/A |
| readonly | String | N/A |
| VendorApiKits | String | N/A |
| CommunityApiKits | String | N/A |
| blog | String | N/A |
| forum | String | N/A |
| support | String | N/A |
| accountReq | String | N/A |
| commercial | String | N/A |
| provider | String | N/A |
| managedBy | String | N/A |
| nonCommercial | String | N/A |
| dataLicensing | String | N/A |
| fees | String | N/A |
| limits | String | N/A |
| terms | String | N/A |
| company | String | N/A |
| updated | Date | N/A |
| Apis | N/A | List of apis further contains apiname and url(as explained above) |

Following is the screenshot of the database data structure for apis records data, in the mongodb shell

```
Command Prompt - mongo.exe

Type "it" for more
> db.apis.findOne()
{
  "_id" : ObjectId<"571961927bebad15099dc004">,
  "id" : "http://www.programmableweb.com/api/the-global-proteome-machine",
  "title" : " The Global Proteome Machine",
  "summary" : "Proteome data for biomedical research",
  "rating" : 4.4000000095367432,
  "name" : " The Global Proteome Machine",
  "label" : " The Global Proteome Machine",
  "author" : null,
  "description" : "The Global Proteome Machine is an attempt to create knowledge from proteomics data and reuse it to solve biomedical research problems. The Global Proteome Machine Database was built to use GPM data to help validate peptide MS/MS spectra and protein coverage patterns. The Global Proteome Machine Database API provides RESTful access to commonly required information based on data from the GPM Database. Responses are JSON formatted.",
  "type" : 1,
  "downloads" : null,
  "useCount" : null,
  "sampleUrl" : "http://wiki.thegpm.org/wiki/GPMDB_REST",
  "downloadUrl" : null,
  "dateModified" : ISODate<"2012-12-17T14:51:40Z">,
  "remoteFeed" : null,
  "numComments" : null,
  "commentsUrl" : "http://api.programmableweb.com/apis/the-global-proteome-machine/comments",
  "Tags" : [
    "science"
  ],
  "category" : "Science",
  "protocols" : "REST",
  "serviceEndpoint" : "http://gpmdb.thegpm.org/",
  "version" : null,
  "wsdl" : null,
  "data Format" : "JSON",
  "apigroups" : null,
  "example" : [
    "database",
    "science"
  ],
  "clientInstall" : null,
  "authentication" : null,
  "ssl" : null,
  "readonly" : null,
  "VendorApiKits" : null,
  "CommunityApiKits" : null,
  "blog" : null,
  "forum" : null,
  "support" : null,
  "accountReq" : "No",
  "commercial" : null,
  "provider" : "http://www.thegpm.org/",
  "managedBy" : null,
  "nonCommercial" : null,
  "dataLicensing" : null,
  "fees" : null,
  "limits" : null,
  "terms" : null,
  "company" : null,
  "updated" : ISODate<"2012-12-17T14:51:40Z">
}
```

Following is the screenshot of the database data structure for mashup data, in the mongodb shell

```
> db.mashup.findOne()
{
  "_id" : ObjectId("571961ef7bebd73c1d3f93b3"),
  "id" : "http://www.programmableweb.com/mashup/-22",
  "title" : null,
  "summary" : null,
  "rating" : 3.7999999952316284,
  "name" : null,
  "label" : null,
  "author" : "Unknown",
  "description" : null,
  "type" : null,
  "downloads" : "0",
  "useCount" : 0,
  "sampleUrl" : "http://www.easypeasyphotos.net",
  "dateModified" : ISODate("2011-10-09T18:35:06Z"),
  "numComments" : 0,
  "commentsUrl" : "http://api.programmableweb.com/mashups/-22/comments",
  "tags" : [
    ""
  ],
  "Apis" : [
    {
      "api" : {
        "apiName" : "Flickr",
        "apiUrl" : "http://www.programmableweb.com/api/flickr"
      }
    },
    {
      "api" : {
        "apiName" : "Google Maps",
        "apiUrl" : "http://www.programmableweb.com/api/google-maps"
      }
    }
  ],
  "updated" : ISODate("2011-10-09T18:35:06Z")
}
```

Testing scenarios

Testing Scenarios for Api Records

When the application is started,

Initially the options to choose are as follows, which are common to both api and mashup search

The screenshot shows a web browser window with the address bar displaying 'localhost:8082/ProgrammableApi/'. The page title is 'Search Programmable Api'. The main heading is 'Programmable Apis/Mashup Search'. Below this, there are two search sections: 'Search By Criteria' and 'Search By Keyword'. In the 'Search By Criteria' section, a red box highlights the 'Select Search' dropdown menu, which currently shows 'Select'. Below this, there are radio buttons for 'Updated Year' (selected) and 'Tags'. A text input field labeled 'Enter Value to Search' contains the value '2013', and a 'Submit' button is below it. In the 'Search By Keyword' section, there is a 'Select Search' dropdown menu showing 'API', a text input field labeled 'Enter Keyword to Search', and a 'Submit' button.

When you select Api in drop down you will get api specific options as follows,

Now let's search for updated year 2013, for that you will have to click on desired radio button to select the criteria which is Updated year as shown below

The screenshot shows a web browser window with the URL `localhost:8082/ProgrammableApi/`. The page title is "Programmable APIs/Mashup Search". Under the heading "Search By Criteria", there is a form with the following elements:

- "Select Search" dropdown menu set to "Api".
- "Select Search Criteria" section with radio buttons for "Updated Year" (selected), "Tags", "Protocols", and "Category".
- "Rating:" dropdown menu set to "Less Than".
- "Enter Value to Search" text input field containing "2013".
- "Submit" button.

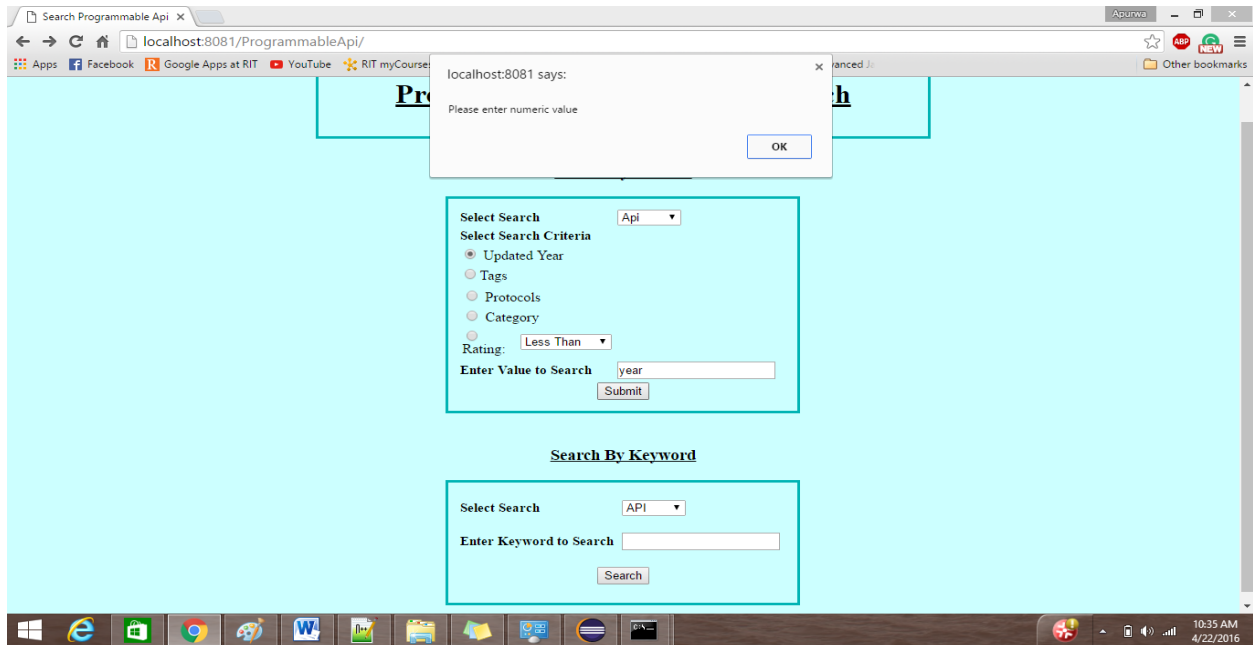
Below this, there is a "Search By Keyword" section with a "Select Search" dropdown menu set to "API" and an "Enter Keyword to Search" text input field, with a "Submit" button.

Results for the updated year 2013 is as follows,

The screenshot shows the same web browser window, but the URL is now `localhost:8082/ProgrammableApi/resultList.jsp`. The page title is "Programmable APIs/Mashup Search". Under the heading "Results for apis", there is a table with the following data:

| Api No | Name | URL |
|--------|------------------------------|---|
| 1 | Atomic Mass Email Service | http://www.programmableweb.com/api/atomic-mass-email-service |
| 2 | Digital Ocean | http://www.programmableweb.com/api/digital-ocean |
| 3 | LocalDiffusion | http://www.programmableweb.com/api/localdiffusion |
| 4 | SupportFu | http://www.programmableweb.com/api/supportfu |
| 5 | Exploring Surrey's past | http://www.programmableweb.com/api/exploring-surreys-past |
| 6 | MarineTraffic | http://www.programmableweb.com/api/marinetraffic |
| 7 | NVMS | http://www.programmableweb.com/api/nvms |
| 8 | iSpot.tv | http://www.programmableweb.com/api/spot.tv |
| 9 | White House Policy Snapshots | http://www.programmableweb.com/api/white-house-policy-snapshots |
| 10 | AddNag | http://www.programmableweb.com/api/addnag |
| 11 | Harvard Dash | http://www.programmableweb.com/api/harvard-dash |
| 12 | FancyBox | http://www.programmableweb.com/api/fancybox |
| 13 | Betterific | http://www.programmableweb.com/api/betterific |
| 14 | Cryptank | http://www.programmableweb.com/api/cryptank |
| 15 | LmBox | http://www.programmableweb.com/api/lmbox |
| 16 | Diffbot Frontpage | http://www.programmableweb.com/api/diffbot-frontpage |

In case the user enters non numeric value for year, the application validates and throws an error
“Please enter numeric value”



Results For Query to search according to the updated year: in the mongodb shell

db.apis.aggregate({\$project: {name: 1, id:2, year: {\$year: '\$updated'}}},{ \$match:{ year: 2013}});

```
Command Prompt - mongo.exe

Type "it" for more
> db.apis.aggregate(<{$project: {name: 1, id:2, year: {$year: '$updated'}}},<$mat
ch: { year: 2013}}>).pretty()
{
  "_id" : ObjectId<"571961927bebad15099dc005">,
  "id" : "http://www.programmableweb.com/api/csc-e-sim",
  "name" : "##CsC e-Sim",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc007">,
  "id" : "http://www.programmableweb.com/api/gah-people",
  "name" : "##Gah People",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc011">,
  "id" : "http://www.programmableweb.com/api/123contactform",
  "name" : "123ContactForm",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc016">,
  "id" : "http://www.programmableweb.com/api/1map",
  "name" : "1Map",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc019">,
  "id" : "http://www.programmableweb.com/api/21-forty-medical-district-sli
de-show-data-service",
  "name" : "21 Forty Medical District Slide Show Data Service",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc01c">,
  "id" : "http://www.programmableweb.com/api/24-pull-requests",
  "name" : "24 Pull Requests",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc01e">,
  "id" : "http://www.programmableweb.com/api/2600hz",
  "name" : "2600hz",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc01f">,
  "id" : "http://www.programmableweb.com/api/27-seconds-knowledge-base",
  "name" : "27 Seconds Knowledge Base",
  "year" : 2013
}
{
  "_id" : ObjectId<"571961927bebad15099dc023">,
  "id" : "http://www.programmableweb.com/api/3.0-trippin-in",
  "name" : "3.0 Trippin&#039; in",
  "year" : 2013
}
```

Search For tags: database select tags as a criteria and enter database in the text field

Search Programmable Api X Apurva

localhost:8082/ProgrammableApi/

Apps Facebook Google Apps at RIT YouTube RIT myCourses RIT Sign In World's Largest Prof: Fcn Is CSCI-605 Advanced J Other bookmarks

Programmable Apis/Mashup Search

Search By Criteria

Select Search

Select Search Criteria

☐ Updated Year

☒ Tags

☐ Protocols

☐ Category

Rating:

Enter Value to Search

Search By Keyword

Select Search

Enter Keyword to Search

The results are

Search Programmable Api X Apurva

localhost:8082/ProgrammableApi/resultList.jsp

Apps Facebook Google Apps at RIT YouTube RIT myCourses RIT Sign In World's Largest Prof: Fcn Is CSCI-605 Advanced J Other bookmarks

Programmable Apis/Mashup Search

Results for apis

| Api No | Name | URL |
|--------|-----------------------------|---|
| 1 | InterMine | http://www.programmableweb.com/api/intermine |
| 2 | RdbHost | http://www.programmableweb.com/api/rdbhost |
| 3 | Evrything | http://www.programmableweb.com/api/evrything |
| 4 | Xeround Cloud Database | http://www.programmableweb.com/api/xeround-cloud-database |
| 5 | Boston Predictive Analytics | http://www.programmableweb.com/api/boston-predictive-analytics |
| 6 | Database.com | http://www.programmableweb.com/api/database.com |
| 7 | FluidDB | http://www.programmableweb.com/api/fluiddb |
| 8 | Datalanche | http://www.programmableweb.com/api/datalanche |
| 9 | NextDB | http://www.programmableweb.com/api/nextdb |
| 10 | gubb | http://www.programmableweb.com/api/gubb |
| 11 | KuroBase | http://www.programmableweb.com/api/kurobase |
| 12 | MetNet | http://www.programmableweb.com/api/metnet |
| 13 | Amazon SimpleDB | http://www.programmableweb.com/api/amazon-simpledb |
| 14 | ThriftDB | http://www.programmableweb.com/api/thriftdb |
| 15 | MyTaskHelper | http://www.programmableweb.com/api/mytaskhelper |
| 16 | TheBigDB | http://www.programmableweb.com/api/thebigdb |
| 17 | M/DB | http://www.programmableweb.com/api/m-db |
| 18 | ClearDB | http://www.programmableweb.com/api/cleardb |

The query to find the database tag in the mongodb:

```
db.apis.find({ "Tags" : { "$regex" : "database" , "$options" : "i" } })
```

Result:



```
Command Prompt - mongo.exe

{"name" : "Xeround Cloud Database",
  "label" : "Xeround Cloud Database",
  "author" : null,
  "description" : "Xeround is a company providing a cloud database SaaS th
at is fully compatible with MySQL and is stored on Xeround&#039;s servers on
Amazon EC2 and RackSpace. The Xeround Cloud Database API is a SOAP or REST API
that allows developers to build their own cloud database management software/app
lication or integrate their cloud management into existing applications. The Xer
ound API is only available to registered users of the Xeround&#039;s cloud d
atabase service.",
  "type" : 1,
  "downloads" : null,
  "useCount" : null,
  "sampleUrl" : "http://xeround.com/",
  "downloadUrl" : null,
  "dateModified" : ISODate("2011-09-23T07:42:03Z"),
  "remoteFeed" : null,
  "numComments" : null,
  "commentsUrl" : "http://api.programmableweb.com/apis/xeround-cloud-datab
ase/comments",
  "Tags" : [
    "database"
  ],
  "category" : "Database",
  "protocols" : "REST, SOAP",
  "serviceEndpoint" : "https://api.xeround.com:8443/1.0/rest",
  "version" : null,
  "wsdl" : "https://api.xeround.com:8443/1.0/soap?wsdl",
  "dataFormat" : "XML",
  "apigroups" : null,
  "example" : [
    "cloud",
    "database"
  ],
  "clientInstall" : null,
  "authentication" : "HTTP Basic Authentication",
  "ssl" : "Yes",
  "readonly" : null,
  "vendorApiKits" : null,
  "communityApiKits" : null,
  "blog" : null,
  "forum" : null,
  "support" : null,
  "accountReq" : "Yes",
  "commercial" : null,
  "provider" : "http://xeround.com/",
  "managedBy" : null,
  "nonCommercial" : null,
  "dataLicensing" : null,
  "fees" : null,
  "limits" : null,
  "terms" : "http://xeround.com/terms-conditions/",
  "company" : null,
  "updated" : ISODate("2011-09-23T07:42:03Z")}
> db.apis.find(<< "Tags" :    { "$regex" : "database" , "$options" : "i" }>>).pretty
<>
```

Search for a protocol for REST, select a search criteria protocols from the radiobutton

The screenshot shows a web browser window with the URL `localhost:8082/ProgrammableApi/`. The page title is "Programmable Apis/Mashup Search". Below the title, there are two search sections. The first section, "Search By Criteria", contains a "Select Search" dropdown menu set to "Api", a "Select Search Criteria" section with radio buttons for "Updated Year", "Tags", "Protocols" (which is selected), and "Category", a "Rating" dropdown set to "Less Than", and an "Enter Value to Search" text box containing "REST". A "Submit" button is located below the text box. The second section, "Search By Keyword", contains a "Select Search" dropdown menu set to "API" and an "Enter Keyword to Search" text box, also with a "Submit" button below it.

Results are as follows

The screenshot shows the same web browser window, but now displaying the "Results for apis" section. This section contains a table with 18 rows of search results. Each row includes an "Api No", a "Name", and a "URL". The URLs are all from `http://www.programmableweb.com/api/`.

| Api No | Name | URL |
|--------|---------------------------|---|
| 1 | Woocation Cityinfo | http://www.programmableweb.com/api/woocation-cityinfo |
| 2 | Atomic Mass Email Service | http://www.programmableweb.com/api/atomic-mass-email-service |
| 3 | 3scale Service Management | http://www.programmableweb.com/api/3scale-service-management |
| 4 | Digital Ocean | http://www.programmableweb.com/api/digital-ocean |
| 5 | Heliod | http://www.programmableweb.com/api/heliod |
| 6 | Chronomove | http://www.programmableweb.com/api/chronomove |
| 7 | FlyMine | http://www.programmableweb.com/api/flymine |
| 8 | VIOLIN | http://www.programmableweb.com/api/violin |
| 9 | Harvard Dash | http://www.programmableweb.com/api/harvard-dash |
| 10 | Radarmatic | http://www.programmableweb.com/api/radarmatic |
| 11 | Bing Traffic | http://www.programmableweb.com/api/bing-traffic |
| 12 | BitTrust | http://www.programmableweb.com/api/bittrust |
| 13 | NHS | http://www.programmableweb.com/api/nhs |
| 14 | Phanfare | http://www.programmableweb.com/api/phanfare |
| 15 | Athlinks | http://www.programmableweb.com/api/athlinks |
| 16 | Realtime Register | http://www.programmableweb.com/api/realtime-register |
| 17 | WallaBee | http://www.programmableweb.com/api/wallabee |
| 18 | directSMS | http://www.programmableweb.com/api/directsms |

Query to find protocols REST in protocols field, in the mongodb shell

```
db.apis.find({ "protocols" : { "$regex" : "REST" , "$options" : "i" } })
```

```
>
Type "it" for more
> db.apis.find({ "protocols" : { "$regex" : "REST" , "$options" : "i" } }).pretty()
>
```

Result:



```
Command Prompt - mongo.exe

{
  "_id" : ObjectId("571961927bebad15099dc022"),
  "id" : "http://www.programmableweb.com/api/2sms",
  "title" : "2sms",
  "summary" : "SMS messaging service",
  "rating" : 3.5,
  "name" : "2sms",
  "label" : "2sms",
  "author" : null,
  "description" : "2sms provides text messaging services for businesses of all sizes. Services include one way and two way messaging, pager replacement, long messaging, reporting and more. 2sms offers an API that allows developers to integrate messaging services into their applications. The API exposes the SMS gateway and is available as in both RESTful and SOAP versions.",
  "type" : 1,
  "downloads" : null,
  "useCount" : null,
  "sampleUrl" : "http://www.2sms.com/software.aspx",
  "downloadUrl" : null,
  "dateModified" : ISODate("2012-09-19T11:37:57Z"),
  "remoteFeed" : null,
  "numComments" : null,
  "commentsUrl" : "http://api.programmableweb.com/apis/2sms/comments",
  "Tags" : [
    "sms"
  ],
  "category" : "Messaging",
  "protocols" : "REST, SOAP",
  "serviceEndpoint" : null,
  "version" : null,
  "wsdl" : null,
  "data Format" : "XML",
  "apigroups" : null,
  "example" : [
    "messaging",
    "sms"
  ],
  "clientInstall" : null,
  "authentication" : null,
  "ssl" : null,
  "readonly" : null,
  "VendorApiKits" : null,
  "CommunityApiKits" : null,
  "blog" : null,
  "forum" : null,
  "support" : null,
  "accountReq" : null,
  "commercial" : null,
  "provider" : "http://www.2sms.com/",
  "managedBy" : null,
  "nonCommercial" : null,
  "dataLicensing" : null,
  "fees" : null,
  "limits" : null,
  "terms" : null,
  "company" : null,
  "updated" : ISODate("2012-09-19T11:37:57Z")
}
```

Search for a category **science** : select criteria category and enter science in the text box

The screenshot shows a web browser window with the URL `localhost:8082/ProgrammableApi/`. The page title is **Programmable Apis/Mashup Search**. Below the title, there are two search sections. The first section, **Search By Criteria**, contains a form with the following elements: a 'Select Search' dropdown menu set to 'Api'; a 'Select Search Criteria' section with radio buttons for 'Updated Year', 'Tags', 'Protocols', 'Category' (which is selected), and 'Rating'; a 'Rating' dropdown menu set to 'Less Than'; an 'Enter Value to Search' text box containing the word 'science'; and a 'Submit' button. The second section, **Search By Keyword**, contains a 'Select Search' dropdown menu set to 'API' and an 'Enter Keyword to Search' text box, also with a 'Submit' button.

Results are

The screenshot shows the same web browser window, but the URL is now `localhost:8082/ProgrammableApi/resultList.jsp`. The page title is still **Programmable Apis/Mashup Search**. Below the title, there is a section titled **Results for apis** containing a table with 15 rows of search results. Each row includes an 'Api No', a 'Name', and a 'URL'.

| Api No | Name | URL |
|--------|--|---|
| 1 | Pseudoviewer | http://www.programmableweb.com/api/pseudoviewer |
| 2 | Mammalian Adult Neurogenesis Gene Ontology | http://www.programmableweb.com/api/mammalian-adult-neurogenesis-gene-ontology |
| 3 | EBI WU-BLAST | http://www.programmableweb.com/api/ebi-wu-blast |
| 4 | Passta | http://www.programmableweb.com/api/passta |
| 5 | QuickGO | http://www.programmableweb.com/api/quickgo |
| 6 | Reactome | http://www.programmableweb.com/api/reactome |
| 7 | Stockholm University PRODIV-TMHMM | http://www.programmableweb.com/api/stockholm-university-prodiv-tmhmm |
| 8 | NREL Solar | http://www.programmableweb.com/api/nrel-solar |
| 9 | DataFed WCS | http://www.programmableweb.com/api/datafed-wcs |
| 10 | GeoAstro | http://www.programmableweb.com/api/geoastro |
| 11 | Chemical Identifier Resolver | http://www.programmableweb.com/api/chemical-identifier-resolver |
| 12 | ChromA | http://www.programmableweb.com/api/chroma |
| 13 | WEHUB | http://www.programmableweb.com/api/wehub |
| 14 | Norwegian Bioinformatics Platform | http://www.programmableweb.com/api/norwegian-bioinformatics-platform |
| 15 | Nunismatic Search | http://www.programmableweb.com/api/nunismatic-search |

Query to search Category , in the mongodb shell

```
>
Type "it" for more
> db.apis.find(<< "category" : { "$regex" : "science" , "$options" : "i" }>>).pretty()
```

Results:



```
Command Prompt - mongo.exe

{"name": "BIOBASE",
"label": "BIOBASE",
"author": null,
"description": "BIOBASE is the leading provider of expert-curated biological databases, software and services for the life sciences. BIOBASE products are used to identify relations that aid in drug and biomarker discovery. BIOBASE helps researchers to identify connections by offering well-structured data, assembled by highly qualified subject-matter experts, organized in an accessible and easily searchable manner. This data is accessible via an API. The API uses RESTful calls and responses are formatted in XML.",
"type": 1,
"downloads": null,
"useCount": null,
"sampleUrl": "http://www.biobase-international.com/solutions-for/bioinformatics",
"downloadUrl": null,
"dateModified": ISODate("2012-06-11T14:36:37Z"),
"remoteFeed": null,
"numComments": null,
"commentsUrl": "http://api.programmableweb.com/apis/biobase/comments",
"Tags": [
  "science"
],
"category": "Science",
"protocols": "REST",
"serviceEndpoint": null,
"version": null,
"wsdl": null,
"data Format": "XML",
"apigroups": null,
"example": [
  "biology",
  "reference",
  "research",
  "science"
],
"clientInstall": null,
"authentication": null,
"ssl": null,
"readonly": null,
"VendorApiKits": null,
"CommunityApiKits": null,
"blog": null,
"forum": null,
"support": null,
"accountReq": null,
"commercial": null,
"provider": "http://www.biobase-international.com/",
"managedBy": null,
"nonCommercial": null,
"dataLicensing": null,
"fees": null,
"limits": null,
"terms": null,
"company": null,
"updated": ISODate("2012-06-11T14:36:37Z")}
```

Now lets search for all the apis having rating less than 3

As you can see when you click on rating you will have to select one of the following option less than, equal to , greater than values and the results will be shown accordingly

Programmable Apis/Mashup Search

Search By Criteria

Select Search:

Select Search Criteria:

- ☐ Updated Year
- ☐ Tags
- ☐ Protocols
- ☐ Category
- ☒ Rating

Rating: **Less Than**

Enter Value:

Search By Keyword

Select Search:

Enter Keyword to Search:

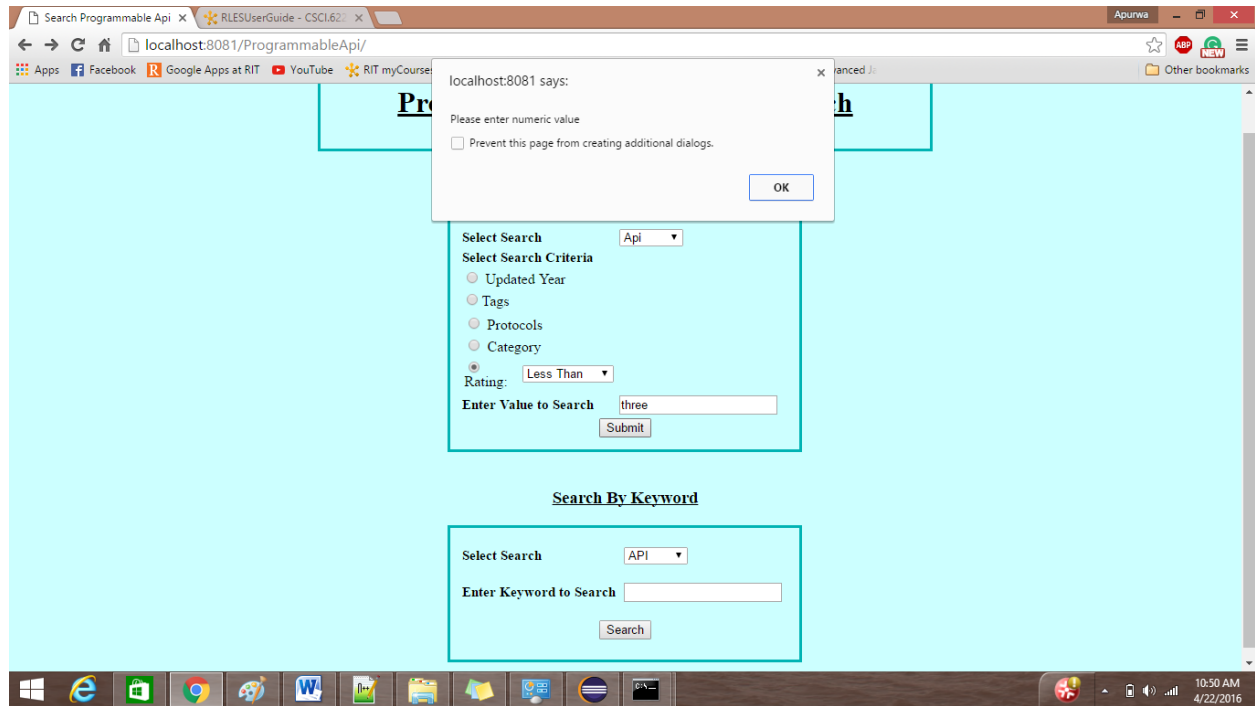
The results are as follows,

Programmable Apis/Mashup Search

Results for apis

| Api No | Name | URL |
|--------|---------------------------|---|
| 1 | Entellium | http://www.programmableweb.com/api/entellium |
| 2 | Rentometer | http://www.programmableweb.com/api/rentometer |
| 3 | Dossia | http://www.programmableweb.com/api/dossia |
| 4 | Vuclip Video | http://www.programmableweb.com/api/vuclip-video |
| 5 | Tagyu | http://www.programmableweb.com/api/tagyu |
| 6 | Beatport | http://www.programmableweb.com/api/beatport |
| 7 | Guerrilla Mail | http://www.programmableweb.com/api/guerrilla-mail |
| 8 | Magicspatula Stock Quotes | http://www.programmableweb.com/api/magicspatula-stock-quotes |
| 9 | MapMyRun | http://www.programmableweb.com/api/mapmyrun |
| 10 | AI Applied Text Extract | http://www.programmableweb.com/api/ai-applied-text-extract |
| 11 | CommonTimes | http://www.programmableweb.com/api/commontimes |
| 12 | USAJOBS | http://www.programmableweb.com/api/usajobs |
| 13 | 360voice | http://www.programmableweb.com/api/360voice |
| 14 | Mapfluence | http://www.programmableweb.com/api/mapfluence |
| 15 | F My Life | http://www.programmableweb.com/api/f-my-life |
| 16 | India PINCode | http://www.programmableweb.com/api/india-pincode |
| 17 | Streamload | http://www.programmableweb.com/api/streamload |
| 18 | Sabre | http://www.programmableweb.com/api/sabre |

The application throws an error if the user enters non numeric value for rating chosen, "Please enter numeric value"



Query to search for api having rating 3 is as follows,

```
db.apis.find({ "rating" : { "$lt" : 3.0}}).pretty()
```

```
>
Type "it" for more
> db.apis.find(<< "rating" : < "$lt" : 3.0}>>).pretty(<>_
```

Results are:

```
Command Prompt - mongo.exe

  "_id" : ObjectId("571961957bebad15099dc54e"),
  "id" : "http://www.programmableweb.com/api/cafe-press",
  "title" : "Cafe Press",
  "summary" : "Customized retail product service",
  "rating" : 2.7000000047683716,
  "name" : "Cafe Press",
  "label" : "Cafe Press",
  "author" : null,
  "description" : "Allows developers to make queries and run searches against Cafe Press&#039; store and product database.",
  "type" : 1,
  "downloads" : null,
  "useCount" : null,
  "sampleUrl" : "http://api.cafepress.com/",
  "downloadUrl" : null,
  "dateModified" : ISODate("2006-03-06T15:02:38Z"),
  "remoteFeed" : null,
  "numComments" : null,
  "commentsUrl" : "http://api.programmableweb.com/apis/cafe-press/comments",
  "Tags" : [
    "shopping"
  ],
  "category" : "Shopping",
  "protocols" : "SOAP, REST",
  "serviceEndpoint" : "http://api.cafepress.com/",
  "version" : null,
  "wsdl" : "http://api.cafepress.com/index.asmx?WSDL",
  "data Format" : "XML",
  "apigroups" : null,
  "example" : [
    "personalization",
    "shopping"
  ],
  "clientInstall" : "No",
  "authentication" : null,
  "ssl" : null,
  "readonly" : "No",
  "VendorApiKits" : null,
  "CommunityApiKits" : null,
  "blog" : null,
  "forum" : null,
  "support" : null,
  "accountReq" : null,
  "commercial" : null,
  "provider" : "http://www.cafepress.com",
  "managedBy" : "Mashery",
  "nonCommercial" : null,
  "dataLicensing" : null,
  "fees" : null,
  "limits" : null,
  "terms" : null,
  "company" : null,
  "updated" : ISODate("2006-03-06T15:02:38Z")
}
Type "it" for more
>
```

For the keyword search enter the keywords in the second form:

Let's enter **"Briteverify restful json"** keywords into the text box as shown below and select API from the dropdown options

The screenshot shows a web browser window with the URL `localhost:8082/ProgrammableApi/`. The page title is **Programmable Apis/Mashup Search**. There are two search forms. The first form, titled **Search By Criteria**, has a dropdown menu for 'Select Search' (set to 'Select'), radio buttons for 'Updated Year' (selected) and 'Tags', and a text input for 'Enter Value to Search'. The second form, titled **Search By Keyword**, is highlighted with a red rectangle. It has a dropdown menu for 'Select Search' (set to 'API') and a text input for 'Enter Keyword to Search' containing the text 'Briteverify restful json'. A 'Search' button is located below the text input.

The results for the same are

The screenshot shows the same web browser window, but the URL is now `localhost:8082/ProgrammableApi/resultList.jsp`. The page title is **Programmable Apis/Mashup Search**. Below the title is the heading **Results for null**. A table is displayed with the following data:

| Api No | Name | URL |
|--------|-------------|---|
| 1 | BriteVerify | http://www.programmableweb.com/api/briteverify |

Query for the keyword search is as follows,

```
db.apis.find({ "$text" : { "$search" : "\"Briteverify\" \"restful\" \"json\""}})
```



```
Command Prompt - mongo.exe
>
Type "it" for more
> db.apis.find(<{ \"$text\" : { \"$search\" : \"\\\"Briteverify\\\" \\\"restful\\\" \\\"json\\\"\"}}>>
.pretty()
{
  "_id" : ObjectId("571961957bebad15099dc4c3"),
  "id" : "http://www.programmableweb.com/api/briteverify",
  "title" : "BriteVerify",
  "summary" : "Realtime email data verification",
  "rating" : 2.90000000953674316,
  "name" : "BriteVerify",
  "label" : "BriteVerify",
  "author" : null,
  "description" : "BriteVerify is an email data verification platform that
connects with email domains to perform real time account status verification of
email addresses. The API allows developers to integrate the system backend with
3rd party applications. The data elements available for verification are: Name,
Email, Postal Address, Phone, IP Address. The API uses RESTful calls and respon
ses are formatted in XML and JSON.",
  "type" : 1,
  "downloads" : null,
  "useCount" : null,
  "sampleUrl" : "https://docs.google.com/file/d/0B0UBKKQ-US7CYmJlMTY0MTQtN
zA0MC00ZjlkLWFlMGQtZmYyN2Y3ZGI5NGMz/edit?authkey=CKSAocMI&hl=en#",
  "downloadUrl" : null,
  "dateModified" : ISODate("2011-07-07T12:26:57Z"),
  "remoteFeed" : null,
  "numComments" : null,
  "commentsUrl" : "http://api.programmableweb.com/apis/briteverify/comment
s",
  "Tags" : [
    "verification"
  ],
  "category" : "Email",
  "protocols" : "REST",
  "serviceEndpoint" : "https://api.briteverify.com/",
  "version" : null,
  "wsdl" : null,
  "data Format" : "XML, JSON",
  "apigroups" : null,
  "example" : [
    "email",
    "realtime",
    "security",
    "verification"
  ],
  "clientInstall" : null,
  "authentication" : "API Key, HTTP Basic Authentication",
  "ssl" : "Yes",
  "readonly" : null,
  "VendorApiKits" : null,
  "CommunityApiKits" : null,
  "blog" : null,
  "forum" : null,
  "support" : null,
  "accountReq" : "Yes",
  "commercial" : null,
}
```

Search For Mashup records:

Select Mashup from the drop down and select updatedyear and enter year value 2013

Programmable Apis/Mashup Search

Search By Criteria

Select Search: Mashup (dropdown menu)
Select Search Criteria: Updated Year (radio button selected), Tags (radio button), Used Apis (radio button)
Enter Value to Search: 2013 (text input field)
Submit (button)

Search By Keyword

Select Search: API (dropdown menu)
Enter Keyword to Search: Briteverify restful json (text input field)
Search (button)

The results are as follows

Programmable Apis/Mashup Search

Results for mashup

| Api No | Name | URL |
|--------|--|---|
| 1 | Find Properly | http://www.programmableweb.com/mashup/find-properly |
| 2 | Kiva Tools | http://www.programmableweb.com/mashup/kiva-tools |
| 3 | Shopinterest | http://www.programmableweb.com/mashup/shopinterest |
| 4 | Park in Toronto | http://www.programmableweb.com/mashup/park-in-toronto |
| 5 | Radar2go | http://www.programmableweb.com/mashup/radar2go |
| 6 | WatchNext | http://www.programmableweb.com/mashup/watchnext |
| 7 | Map of the Dead | http://www.programmableweb.com/mashup/map-of-the-dead |
| 8 | Pennsylvania Wines | http://www.programmableweb.com/mashup/pennsylvania-wines |
| 9 | Bluth Radio | http://www.programmableweb.com/mashup/bluth-radio |
| 10 | EDU Libs | http://www.programmableweb.com/mashup/edu-libs |
| 11 | Price Drop You Save | http://www.programmableweb.com/mashup/price-drop-you-save |
| 12 | DocuSign & BlueWolf | http://www.programmableweb.com/mashup/docusign-bluewolf |
| 13 | Sweet Soundtrack | http://www.programmableweb.com/mashup/sweet-soundtrack |
| 14 | Medical Fundraising, Crisis & Disaster Map | http://www.programmableweb.com/mashup/medical-fundraising-crisis-disaster-map |
| 15 | StoreSlider Deutschland | http://www.programmableweb.com/mashup/storeslider-deutschland |
| 16 | Movius | http://www.programmableweb.com/mashup/movius |
| 17 | DocuSign for Google Drive | http://www.programmableweb.com/mashup/docusign-for-google-drive |

The Query for searching api record according to updated year 2013 is

```
db.mashup.aggregate({$project: {name: 1, id:2, year: {$year: '$updated'}}},{ $match:{ year: 2013}})
```

```
Type "it" for more
> db.mashup.aggregate(<{$project: {name: 1, id:2, year: {$year: '$updated'}}},<$match:{ year: 2013}}).pretty()
```

Results are as follows,



```
Command Prompt - mongo.exe

{"_id" : ObjectId("571961ef7bebd73c1d3f93b5"),
  "id" : "http://www.programmableweb.com/mashup/mobile-emulator",
  "name" : " Mobile Emulator",
  "year" : 2013
}
{
  "_id" : ObjectId("571961ef7bebd73c1d3f93b9"),
  "id" : "http://www.programmableweb.com/mashup/beermap-the-top-2500-beers-on-twitter",
  "name" : "#BeerMap - The Top 2,500 Beers on Twitter",
  "year" : 2013
}
{
  "_id" : ObjectId("571961ef7bebd73c1d3f93bd"),
  "id" : "http://www.programmableweb.com/mashup/s42at",
  "name" : "#S42AT",
  "year" : 2013
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f93c7"),
  "id" : "http://www.programmableweb.com/mashup/100-destinations",
  "name" : "100 Destinations",
  "year" : 2013
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f9402"),
  "id" : "http://www.programmableweb.com/mashup/4wheelz-routemate",
  "name" : "4Wheelz RouteMate",
  "year" : 2013
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f9417"),
  "id" : "http://www.programmableweb.com/mashup/aanbieding",
  "name" : "aanbieding",
  "year" : 2013
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f941a"),
  "id" : "http://www.programmableweb.com/mashup/about-pune-city",
  "name" : "About Pune City",
  "year" : 2013
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f941d"),
  "id" : "http://www.programmableweb.com/mashup/academic-ranking-of-world-universities-shanghai-ranking",
  "name" : "Academic Ranking of World Universities (Shanghai Ranking)",
  "year" : 2013
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f9454"),
  "id" : "http://www.programmableweb.com/mashup/airlift-docuSign",
  "name" : "Airlift & DocuSign",
  "year" : 2013
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f9469"),
```

Search For tags: select Tags as the criteria and enter text in the text box

Search Programmable Api x mongodb text search and x MongoDB Text Search All x \$text — MongoDB Manu x Facebook x Apurwa

localhost:8082/ProgrammableApi/

Programmable Apis/Mashup Search

Search By Criteria

Select Search Mashup

Select Search Criteria

☐ Updated Year

☒ Tags

☐ Used Apis

Enter Value to Search travel

Submit

Search By Keyword

Select Search API

Enter Keyword to Search Briteverify restful json

Search

The results are

Search Programmable Api x mongodb text search and x MongoDB Text Search All x \$text — MongoDB Manu x Facebook x Apurwa

localhost:8082/ProgrammableApi/resultList.jsp

Programmable Apis/Mashup Search

Results for mashup

| Api No | Name | URL |
|--------|-----------------------------------|---|
| 1 | vivirama | http://www.programmableweb.com/mashup/vivirama |
| 2 | Shanklin Explorer | http://www.programmableweb.com/mashup/shanklin-explorer |
| 3 | Motel 6 Attraction Finder | http://www.programmableweb.com/mashup/motel-6-attraction-finder |
| 4 | Coast Radar | http://www.programmableweb.com/mashup/coast-radar |
| 5 | OnTheWay | http://www.programmableweb.com/mashup/ontheway |
| 6 | Toronto Attractions Map | http://www.programmableweb.com/mashup/toronto-attractions-map |
| 7 | Track Flight Status | http://www.programmableweb.com/mashup/track-flight-status |
| 8 | Mileage Calculator | http://www.programmableweb.com/mashup/mileage-calculator |
| 9 | zangoa | http://www.programmableweb.com/mashup/zangoa |
| 10 | Map My Hindu Temple | http://www.programmableweb.com/mashup/map-my-hindu-temple |
| 11 | Personal Interactive Travel Agent | http://www.programmableweb.com/mashup/personal-interactive-travel-agent |
| 12 | BelizeMapia | http://www.programmableweb.com/mashup/belizemapia |
| 13 | Kayak Mobile | http://www.programmableweb.com/mashup/kayak-mobile |
| 14 | San Jose Gas Prices | http://www.programmableweb.com/mashup/san-jose-gas-prices |

www.programmableweb.com/mashup/toronto-attractions-map

Hotel <http://www.programmableweb.com/mashup/see-your-hotel>

Query to search records according to the travel as a tag is

```
db.mashup.find({ tags: { $in: [ "travel" ] } })
```

```
>
Type "it" for more
> db.mashup.find({ tags: { $in: [ "travel" ] } }).pretty()
```

Results are

```
Command Prompt - mongo.exe

    "politics",
    "science",
    "travel"
  ],
  "apis" : [
    {
      "api" : {
        "apiName" : "Google Maps",
        "apiUrl" : "http://www.programmableweb.com/api/g
oogle-maps"
      }
    },
    {
      "updated" : ISODate("2006-07-19T01:18:15Z")
    }
  ],
  "_id" : ObjectId("571961f07bebd73cid3f9453"),
  "id" : "http://www.programmableweb.com/mashup/aircraft-flight-tracking-d
emo",
  "title" : "Aircraft Flight Tracking Demo",
  "summary" : "Demonstration that tracks a private aircraft anywhere in th
e world via Iridium satellite network.",
  "rating" : 3.09999999046325684,
  "name" : "Aircraft Flight Tracking Demo",
  "label" : "Aircraft Flight Tracking Demo",
  "author" : "Unknown",
  "description" : "Demonstration that tracks a private aircraft anywhere i
n the world via Iridium satellite network.",
  "type" : null,
  "downloads" : null,
  "useCount" : 6206,
  "sampleUrl" : "http://www.fsinsider.com/SpecialFeatures/BarringtonIrving
/barrington_map_route.aspx",
  "dateModified" : ISODate("2007-04-11T04:45:18Z"),
  "numComments" : 0,
  "commentsUrl" : "http://api.programmableweb.com/mashups/aircraft-flight-
tracking-demo/comments",
  "tags" : [
    "aviation",
    "deadpool",
    "gps",
    "mapping",
    "travel"
  ],
  "apis" : [
    {
      "api" : {
        "apiName" : "Microsoft Bing Maps",
        "apiUrl" : "http://www.programmableweb.com/api/m
icrosoft-bing-maps"
      }
    },
    {
      "updated" : ISODate("2007-04-11T04:45:18Z")
    }
  ],
Type "it" for more
>
```


Search based on Used apis select Used Api as the criteria and enter name of the api in the text

Search Programmable Api x mongodb text search and x MongoDB Text Search All x \$text — MongoDB Manu x Facebook x Apurwa

localhost:8082/ProgrammableApi/

Programmable Apis/Mashup Search

Search By Criteria

Select Search Mashup

Select Search Criteria

☐ Updated Year

☐ Tags

☒ Used Apis

Enter Value to Search google maps

Submit

Search By Keyword

Select Search API

Enter Keyword to Search Briteverify restful json

Search

Results

Search Programmable Api x mongodb text search and x MongoDB Text Search All x \$text — MongoDB Manu x Facebook x Apurwa

localhost:8082/ProgrammableApi/resultList.jsp

Programmable Apis/Mashup Search

Results for mashup

| Api No | Name | URL |
|--------|--|---|
| 1 | BerkeleyCA crimelog.org | http://www.programmableweb.com/mashup/berkeleyca-crimelog.org |
| 2 | Blosh | http://www.programmableweb.com/mashup/blosh |
| 3 | mapmap.cz | http://www.programmableweb.com/mashup/mapmap.cz |
| 4 | RunningFree | http://www.programmableweb.com/mashup/runningfree |
| 5 | PlacesData.com | http://www.programmableweb.com/mashup/placesdata.com |
| 6 | Incident1 | http://www.programmableweb.com/mashup/incident1 |
| 7 | Madoff Victim Map | http://www.programmableweb.com/mashup/madoff-victim-map |
| 8 | College Programs Map | http://www.programmableweb.com/mashup/college-programs-map |
| 9 | Western Hemisphere Shorebird Reserves | http://www.programmableweb.com/mashup/western-hemisphere-shorebird-reserves |
| 10 | DoAt (do@) | http://www.programmableweb.com/mashup/doat-do |
| 11 | Track Flight Status | http://www.programmableweb.com/mashup/track-flight-status |
| 12 | Explore California | http://www.programmableweb.com/mashup/explore-california |
| 13 | All of world national anthems mashup service | http://www.programmableweb.com/mashup/all-of-world-national-anthems-mashup-service |
| 14 | RouteSlip | http://www.programmableweb.com/mashup/routeslip |
| 15 | Sky Map | http://www.programmableweb.com/mashup/sky-map |
| 16 | Info Balloon | http://www.programmableweb.com/mashup/info-balloon |

www.programmableweb.com/mashup/incident1

Query to search records having apiname as google maps is as follows,

```
db.mashup.find({"Apis.api.apiName": {"$regex": "google maps", "$options": "i"}}).
```

```
Type "it" for more
> db.mashup.find(<{ "Apis.api.apiName" : { "$regex" : "google maps" , "$options":
  "i" } }>).pretty()
```

The results are as follows,

```
Command Prompt - mongo.exe

    "api" : {
      "apiName" : "Google Maps",
      "apiUrl" : "http://www.programmableweb.com/api/g
oogle-maps"
    }
  },
  "updated" : ISODate("2009-12-11T05:04:00Z")
}
{
  "_id" : ObjectId("571961f07bebd73c1d3f93e0"),
  "id" : "http://www.programmableweb.com/mashup/29-travels",
  "title" : "29 Travels",
  "summary" : "Used to highlight countries you have visited. Supports Goog
le Maps and Google Earth with a browser plugin. Provides an embeddable HTML code
so bloggers and website owners can embed it into their site. No need to registe
r.",
  "rating" : 4.4000000095367432,
  "name" : "29 Travels",
  "label" : "29 Travels",
  "author" : "Unknown",
  "description" : "Used to highlight countries you have visited. Supports
Google Maps and Google Earth with a browser plugin. Provides an embeddable HTML
code so bloggers and website owners can embed it into their site. No need to reg
ister.",
  "type" : null,
  "downloads" : "0",
  "useCount" : 4859,
  "sampleUrl" : "http://www.29travels.com",
  "dateModified" : ISODate("2008-11-25T11:05:47Z"),
  "numComments" : 0,
  "commentsUrl" : "http://api.programmableweb.com/mashups/29-travels/comme
nts",
  "tags" : [
    "mapping",
    "travel"
  ],
  "Apis" : [
    {
      "api" : {
        "apiName" : "Google Earth",
        "apiUrl" : "http://www.programmableweb.com/api/g
oogle-earth"
      }
    },
    {
      "api" : {
        "apiName" : "Google Maps",
        "apiUrl" : "http://www.programmableweb.com/api/g
oogle-maps"
      }
    }
  ],
  "updated" : ISODate("2008-11-25T11:05:47Z")
}
Type "it" for more
> _
```

Keyword search in mashup: select Mashup in the dropdown options and enter keyword as unsafe cities america

The screenshot shows a web browser window with the URL `localhost:8082/ProgrammableApi/`. The page title is **Programmable Apis/Mashup Search**. There are two main search sections:

- Search By Criteria:** Includes a dropdown menu set to 'Mashup', radio buttons for 'Updated Year' and 'Tags', a text input field for 'Enter Value to Search', and a 'Submit' button.
- Search By Keyword:** This section is highlighted with a red box. It includes a dropdown menu set to 'Mashup', a text input field containing 'unsafe cities america', and a 'Search' button.

Results are as follows

The screenshot shows the same web browser window, but the URL is `localhost:8082/ProgrammableApi/resultList.jsp`. The page title is **Programmable Apis/Mashup Search**. Below the title, the text **Results for null** is displayed. A table with one row of results is shown:

| Api No | Name | URL |
|--------|-----------------------|---|
| 1 | 25 Unsafest US Cities | http://www.programmableweb.com/mashup/25-unsafest-us-cities |

```

"id" : Object<Id<"571961f07bebd73c1d3f93de">>,
"id" : "http://www.programmableweb.com/mashup/25-unsafest-us-cities",
"title" : "25 Unsafest US Cities",
"summary" : "A look at the 25 most unsafe cities in America.",
"rating" : 3.09999999046325684,
"name" : "25 Unsafest US Cities",
"label" : "25 Unsafest US Cities",
"author" : "Unknown",
"description" : "A look at the 25 most unsafe cities in America.",
"type" : null,
"downloads" : null,
"useCount" : 24776,
"sampleUrl" : "http://www.mibazaar.com/unsafecities/",
"dateModified" : ISODate<"2006-11-07T06:43:44Z">,
"numComments" : 1,
"commentsUrl" : "http://api.programmableweb.com/mashups/25-unsafest-us-cities/comments",
"tags" : [
  "crime",
  "mapping",
  "trivia"
],
"apis" : [
  {
    "api" : {
      "apiName" : "Google Maps",
      "apiUrl" : "http://www.programmableweb.com/api/google-maps"
    }
  }
],
"updated" : ISODate<"2006-11-07T06:43:44Z">

```