
Writing a Google API Client

Accessing the Popular Search Service
Programmatically

Developed by: **transentia**

Unit 1, 40 Skew St.
Sherwood
Queensland 4075
Australia

Mobile: 0416 078 813
Phone/FAX: (07) 3278 5890
Email: contact@transentia.com.au
WWW: www.transentia.com.au

ACN: 091-142-880
ABN: 680-911-428-80

Exercise

3

Writing a Google API Client

Accessing the Popular Search Service Programmatically

Introduction

Google (<http://www.google.com>), arguably the #1 internet “search engine” has recently made available for general use a simple ‘beta’ WebServices interface.

Simple Java Client

This exercise shows how to write a simple client application in Java that utilises the Google API.

Install the Appropriate Google API Files

In this session you will need:

- The Java SDK
- The Google API, packaged as googleapi.jar
- The Google WSDL file

The appropriate files will be provided.

Create the Client

Execute the following sequence of commands within a Command Prompt window:

```
> mkdir C:\google  
> cd /d C:\google
```

Create the following in the file **GoogleSearcher.java**:

```
import com.google.soap.search.*;  
  
public class GoogleSearcher  
{  
    private static final String  
        QSTRING = "Transentia -filetype:PDF",
```

```

ACCESSKEY = "/k/1+ndImTMpS6P0GrY92mqTp+oKW4ON";

public static void main(String[] args)
{
    try
    {
        GoogleSearch s = new GoogleSearch();
        s.setKey(ACCESSKEY);
        s.setQueryString(QSTRING);
        GoogleSearchResult r = s.doSearch();
        System.out.println("Google Search Results. Term '" +
                           QSTRING + "' was found in:");
        GoogleSearchResultElement [] res = r.getResultElements();
        for (int i = r.getStartIndex() - 1;
             i <= r.getEndIndex() - 1; i++)
            System.out.println (i + ": " + res[i].getURL() +
                               " (" + res[i].getTitle() + ")");
    }
    catch (GoogleSearchFault f)
    {
        System.err.println
            ("The call to the Google Web APIs failed:");
        System.err.println(f.toString());
    }
}
}

```

Points of interest:

- The Google access key, which must be obtained prior to execution and which must be issued for each client request that is made
- The Google API utilises the file GoogleSearch.wsdl, which should be placed in the directory from which the application is executed. It is probably a good idea to take a *quick* look at this file.

Compile the Client

Compilation is simple. Issue this command within the Command Prompt window:

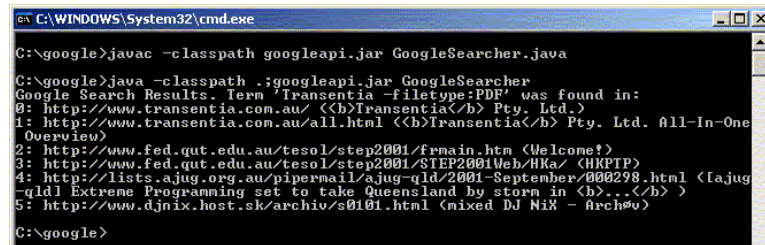
```
C:\google> javac -classpath googleapi.jar GoogleSearcher.java
```

Execution Time!

Execution is also simple:

```
C:\google> java -classpath .;googleapi.jar GoogleSearcher
```

You should be rewarded with a list of all the sites that contain any non PDF documents that refer to 'Transentia', as shown here:



```
C:\WINDOWS\System32\cmd.exe
C:\google>javac -classpath googleapi.jar GoogleSearcher.java
C:\google>java -classpath .;googleapi.jar GoogleSearcher
Google Search Results. Term 'Transentia -filetype:PDF' was found in:
0: http://www.transentia.com.au/ <b>Transentia</b> Pty. Ltd.)
1: http://www.transentia.com.au/all.html <b>Transentia</b> Pty. Ltd. All-In-One
  Overview)
2: http://www.fed.qut.edu.au/tesol/step2001/frmain.htm <Welcome!>
3: http://www.fed.qut.edu.au/tesol/step2001/STEP2001Web/HKa/ <HKIP>
4: http://lists.ajug.org.au/pipermail/ajug-qld/2001-September/000298.html <[ajug
  -qld] Extreme Programming set to take Queensland by storm in <b>...</b> >
5: http://www.djunix.host.sk/archiv/s0101.html <mixed DJ NiK - Archiv>
C:\google>
```

You can verify the returned information by pointing your browser to <http://www.google.co.nz/search?q=Transentia+filetype:PDF>.