Geo-Genealogy Analytics Tool

Instructions For Use

Ross Maciejewski

Jose Ibarra

Arizona State University

Tempe, AZ

Table of Contents

[Instructions to Install Web Service Server on Computer 3](#_Toc370828259)

[Download and Install Eclipse 3](#_Toc370828260)

[Download and Install Tomcat 7.0 3](#_Toc370828261)

[Create Code Folder 3](#_Toc370828262)

[Setup the Eclipse Project 3](#_Toc370828263)

[Download Jersey/MySQL Files and Create Libs Folder 3](#_Toc370828264)

[Copy Jersey Files to Project 4](#_Toc370828265)

[Create Tomcat Server in Eclipse and Copy Needed Files 4](#_Toc370828266)

[Setup Server for Correct Database 5](#_Toc370828267)

[Setup Different Database Type (Non-MySQL) 5](#_Toc370828268)

[Setup Web Application 6](#_Toc370828269)

[Download and Copy the Required Libraries 6](#_Toc370828270)

[Running the Web Application 6](#_Toc370828271)

[Change the Map Location 6](#_Toc370828272)

[Change the Web Service Location 6](#_Toc370828273)

[Additional Instructions 7](#_Toc370828274)

[Running the Web Server 7](#_Toc370828275)

[Change the Columns/Database/Table Searched in Queries 7](#_Toc370828276)

[Remove Limits on SQL Queries 7](#_Toc370828277)

[Screenshots 8](#_Toc370828278)

[Sample Output for Ross Maciejewski 8](#_Toc370828279)

[Sample Output for Jose Ibarra 9](#_Toc370828280)

# Instructions to Install Web Service Server on Computer

## Download and Install Eclipse

1. Navigate to <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/keplerr>
2. Select the relevant Eclipse for your operating system and download it.
3. Install Eclipse
   1. Default settings are fine

## Download and Install Tomcat 7.0

1. Navigate to <http://apache.tradebit.com/pub/tomcat/tomcat-7/v7.0.42/bin/apache-tomcat-7.0.42.exe>
2. After downloaded, install Tomcat
3. Default settings are fine (install should be in C:/ directory)

## Create Code Folder

1. Create a folder to store the code in an easy to find location (i.e. Documents/Code/)
2. Copy the code provided to this folder
3. Load Eclipse. The workspace should be a directory where the coe is stored (i.e. Documents/Code/)
4. If errors appear, that is okay.

## Setup the Eclipse Project

1. Right click the Project Explorer in Eclipse and select Import > Import...
2. Type in "existing projects" in the search bar and select Existing Projects into Workspace
3. Press next then browse for the directory where the code is stored
4. The project "webservice" should be located under the Projects list when the code directory is found.
5. Select the webservice project and Finish

## Download Jersey/MySQL Files and Create Libs Folder

1. Download Jersey 1.17.1 from the following links:
   1. <http://repo1.maven.org/maven2/com/sun/jersey/jersey-archive/1.17.1/jersey-archive-1.17.1.zip>
   2. <http://repo1.maven.org/maven2/com/sun/jersey/jersey-bundle/1.17.1/jersey-bundle-1.17.1.jar>
2. Download Jersey 2.0 RI from the following link:
   1. <http://repo1.maven.org/maven2/org/glassfish/jersey/bundles/jaxrs-ri/2.2/jaxrs-ri-2.2.zip>
3. Download MySQL Connector from the following link (requires registration):
   1. <http://dev.mysql.com/downloads/mirror.php?id=414247>
4. Extract all zip files.
5. Create a folder for libraries called libs (if it doesn't exist) in the web service code location (ie Documents/Code/Facebook Code/webservice/libs)
6. Copy over the entire \lib\ folder from the jersey-archive-1.17.1/lib/ folder (from the Jersey 1.17.1 zip download)
7. Copy over the jersey-bundle-1.17.1.jar that was downloaded (from the Jersey 1.17.1 jar download)
8. Copy over \jaxrs-ri\ext\javax.servlet-api-3.0.1.jar that was downloaded (from the Jersey 2.0 RI download)
9. Copy over the mysql-connector-java-5.1.26-bin.jar that was downloaded (from the MySQL Connector download)
10. Right click the Project Explorer and select Refresh.

## Copy Jersey Files to Project

1. If errors did not disappear after completing the last step of the Download Jersey/MySQL Files and Create Libs Folder section, complete this step. Otherwise, skip this step.
2. Right click the webservice project in Eclipse and select Properties
3. Go to Java Build Path and the Libraries tab
4. Select Add JARs...
5. Navigate to the libs folder location
6. Add the javax.servlet-api-3.0.1.jar file to the project
7. Repeat and add the jersey-bundle.1.17.1.jar file to the project
8. Remove missing .jar files that may be left over from other setups (Missing .jar files have a red x and a (missing) label)

## Create Tomcat Server in Eclipse and Copy Needed Files

1. Right click on the Server tab at the bottom and select New -> Server
2. Select Tomcat v7.0 Server and leave the default host name (localhost)
3. If asked for the Tomcat installation directory, find where the Tomcat folder was installed using the Browse... button (this is typically in the C:/ base folder)
4. Add the webservice project to the projects the server is configured for and press Finish
5. Copy to webservice/WebRoot/WEB-INF/lib/ from webservice/libs/:
   1. mysql-connector-java-5.1.26-bin.jar
   2. asm-3.1.jar
   3. jackson-mapper-asl-1.9.2.jar
   4. javax.servlet-api-3.0.1.jar
   5. jersey-bundle.1.17.1.jar
   6. jackson-core-asl-1.9.2.jar
   7. jackson-jaxrs-1.9.2.jar
   8. jackson-xc-1.9.2.jar
6. Double click the server to open up the overview. Press "Open launch configuration" link under General Information
7. Under the Classpath tab, click the User Entries list then press Add JARs...
8. Add (from the webservice/WebRoot/WEB-INF/lib/ path)
   1. asm-3.1.jar
   2. jackson-mapper-asl-1.9.2.jar
   3. javax.servlet-api-3.0.1.jar
   4. jersey-bundle.1.17.1.jar
   5. jackson-core-asl-1.9.2.jar
   6. jackson-jaxrs-1.9.2.jar
   7. jackson-xc-1.9.2.jar

## Setup Server for Correct Database

1. Open the server configuration folder in Eclipse (typically under Servers/Tomcat\_v7.0\_Server at localhost-config)
2. Open the file context.xml (Right click -> Open With -> Text Editor for easy editing)
3. Add the following lines before the </Context> line:

<Resource name="jdbc/phonebook" auth="Container" type="javax.sql.DataSource"

maxActive="100" maxIdle="30" maxWait="10000"

url="jdbc:mysql://DB\_LOCATION"

driverClassName="com.mysql.jdbc.Driver"

username="USER" password="PASSWORD" />

* 1. DB\_LOCATION is the location of the database (default is localhost:3306/phonebook)
  2. USER is the username to access the database
  3. PASSWORD is the password for the username used to access the database

1. OPTIONAL: If the web service server is not using Tomcat:
   1. Open WFQuery.java in the Java Resources (src/edu.asu.wangfeng.surname.service.resources)
   2. Comment out the uncommented 4 lines of code at the top
   3. Uncomment the lines below //OPTIONALLY: Comment above code and uncomment below code comment
   4. Change the url, username and password strings to the correct values

## Setup Different Database Type (Non-MySQL)

1. Download a new database connector for your server type.
   1. The connector downloaded for this tutorial was for MySQL, from <http://dev.mysql.com/downloads/mirror.php?id=414247>.
   2. Search for the correct .jar connector file for the database type
2. Replace the mysql-connector-java-5.1.26-bin.jar file installed earlier with the new .jar connector file downloaded.
3. In Setup Server for Correct Database instructions:
   1. For a server using Tomcat on the local computer:
      1. Change the line driverClassName="com.mysql.jdbc.Driver" to the correct driver type in context.xml
      2. Change the mysql part of url="jdbc:mysql://DB\_LOCATION" in the context.xml to the correct database type
   2. For servers not using Tomcat (the optional instructions):
      1. Change the String url = "jdbc:mysql://localhost:3306/phonebook"; line in the now uncommented code from mysql to the correct database type.

# Setup Image Folder

1. If given a NullPointerException when creating/capturing an image, the following steps must be followed.
2. Find the base server directory.
   1. This varies on server configuration but in Eclipse with Tomcat, the default is C:\Users\jlibarr1\Documents\Code\Facebook Code\.metadata\.plugins\org.eclipse.wst.server.core\tmp0\wtpwebapps\
3. Within the base server directory, follow the path: webservice/image/
   1. You should now be in the location for images within the server
4. Add a folder named “upload.”

**At this point, everything should be loaded properly for the web service to be hosted on this computer.**

# Setup Web Application

## Download and Copy the Required Libraries

1. Download jQuery UI: <http://jqueryui.com/resources/download/jquery-ui-1.10.2.zip>
2. Unzip jQuery UI folder.
3. Copy the unzipped jQuery UI folder to the lib folder in \Documents\Code\Facebook Code\Web\php\lib

## Running the Web Application

1. Find the index.html file in the provided files (Documents\Code\Facebook Code\Web\html\index.html)
2. Right click it and select Open With > and select your browser of choice.
3. If the server is setup correctly and the database is pointed to correctly, the map should update when a name is entered and submitted.

## Change the Map Location

1. Find the state.js file in the provided files (\Web\php\javascript\) and open it.
2. The centerLat and centerLon variable correspond to the lat and lon of the center of the map
3. zoomLevel corresponds to teh zoom level of the map

## Change the Web Service Location

1. Find the state.js file in the provided files (\Web\php\javascript\) and open it.
2. If the web service server is not ran on this computer with default settings, the serviceBase and httpServiceBase must be changed to the location of the web service server.
3. If the server is hosted on the current computer, the default location of the service is <http://localhost:8080/surnameservice/>

# Additional Instructions

## Running the Web Server

1. With Eclipse open with the settings above, go to the Servers tab at the bottom part of the screen.
   1. Right click the Tomcat server and select start (or restart if already running)
2. Any changes to the web services or server (including the changes above) requires the web server to be restarted.
3. To check the service is running, go to the url of the server and add /surnameservice/ - A webpage should display indicating the service was loaded correctly.
   1. Default location: http://localhost:8080/surnameservice/
   2. Nondefault: <http://WEB_SERVICE_SERVER/surnameservice/>

## Change the Columns/Database/Table Searched in Queries

1. Under the folder webservice/edu.asu.joseibarra.services.name is the QueryName.java file, containing the SQL queriesThe ForenameMapBuilder.java and
2. Two SQL calls are made, both labeled by comments SQL LOCATED HERE
   1. The first SQL gets the latitude and logitude for where people are located, given their surname/forename and a min/max lon/lat.
   2. The second SQL query gets the average mean and average median for a given surname/forename.
   3. The line labeled "CAN BE COMMENTED OUT TO IGNORE INCOME" can be commented out to ignore this query. The following two lines have to be uncommented, however.

## Remove Limits on SQL Queries

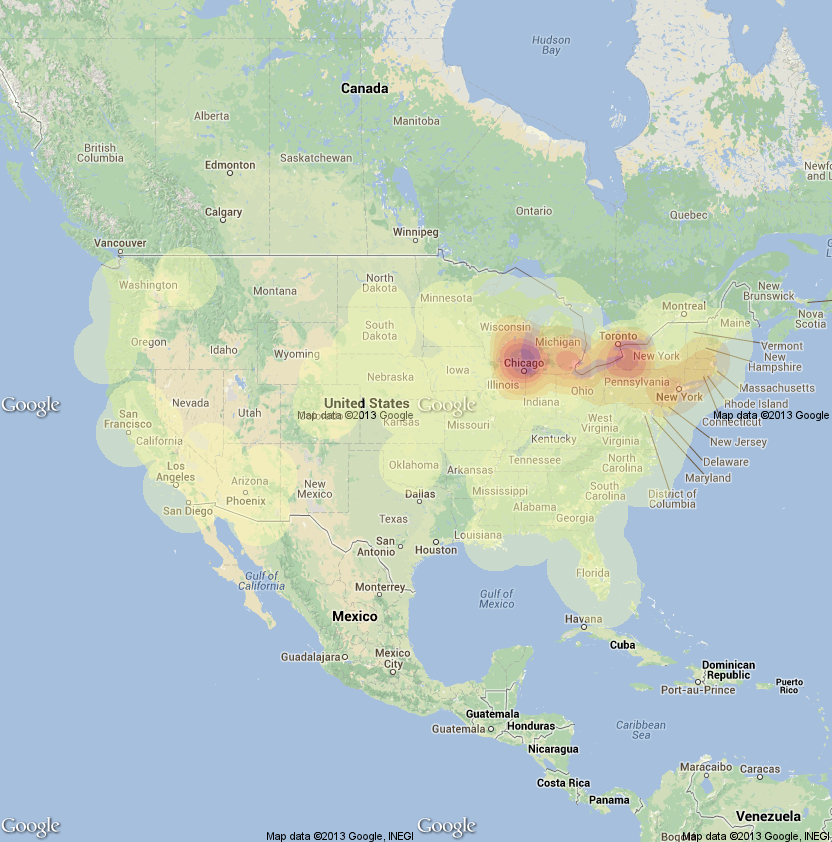
1. Under the folder webservice/edu.asu.joseibarra.services.name is the QueryName.java file, containing the SQL queries and the limit
2. The comment SQL LIMIT HERE indicates the variable used to limit the SQL queries
   1. Change the value of this variable to the max value required
   2. The default limit is 25,000

# Screenshots

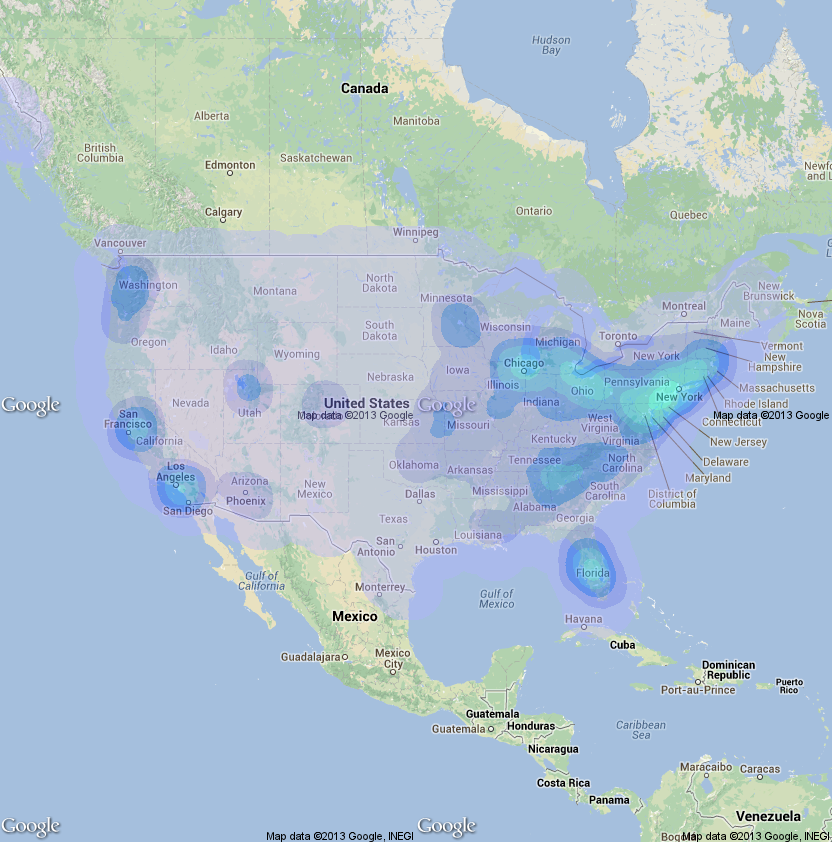
The following are limited queries (25,000 max) of selected names.

## Sample Output for Ross Maciejewski

For surname “Maciejewski,” average mean income $72,136.14 and average median income $57,613.66:

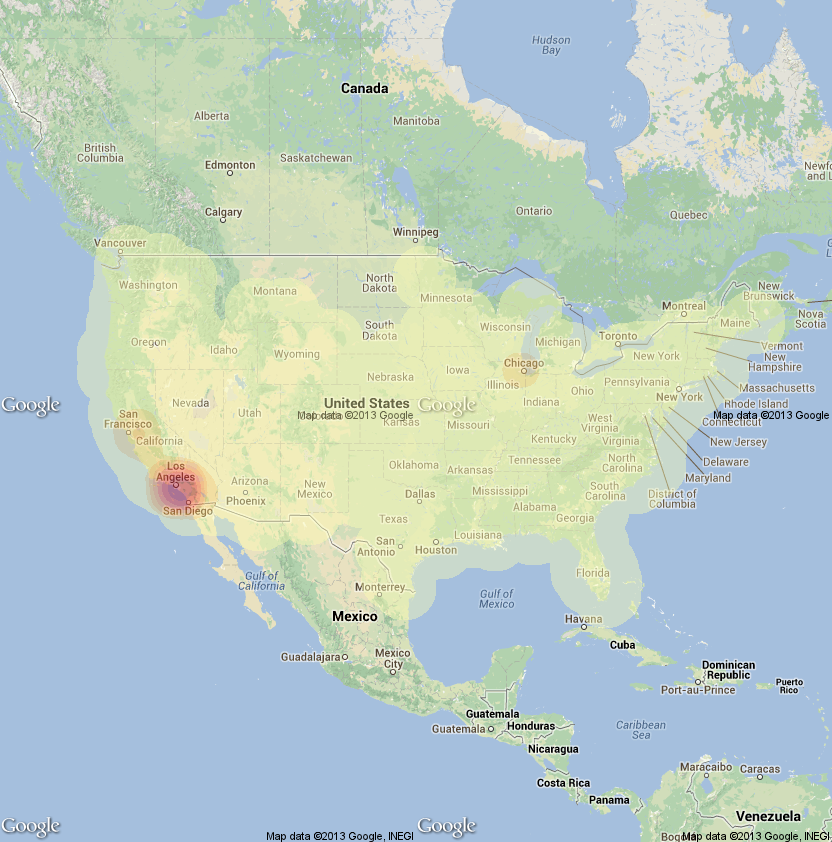


For forename “Ross,” average mean income $75,853.07 and average median income $59,975.18:



## Sample Output for Jose Ibarra

For surname “Ibarra,” average mean income $70,788.16 and average median income $56,231.80:



For forename “Jose,” average mean income $70,787.05 and average median income $55,950.45:

