

```

package android.cohort;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class TwitterNameActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.twitter_name_activity);

        Button searchButton = (Button) findViewById(R.id.search);
        searchButton.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                //Retrieve the search string
                EditText searchKey = (EditText)
                    findViewById(R.id.search_text);
                String searchValue = searchKey.getText().toString().trim();

                //Checks to see if the entry is valid
                if (searchValue.equals("")){
                    String alert =
                        getString(R.string.empty_string_alert);
                    Toast.makeText(getApplicationContext(), alert,
                        Toast.LENGTH_SHORT).show();
                    return;
                }

                //Launches the next screen
                Intent settingsActivity = new
                    Intent(getApplicationContext(),
                        TwitterSearchResultsWithImagesActivity.class);

                //Passes the search string to the next activity
                String extraName = getString(R.string.search_extra_name);
                settingsActivity.putExtra(extraName, searchValue);

                startActivity(settingsActivity);
            }
        });
    }
}

```

```

package android.cohort;

import java.io.IOException;
import java.io.InputStream;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.entity.BufferedHttpEntity;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.params.BasicHttpParams;
import org.apache.http.params.HttpParams;
import org.json.JSONException;
import org.json.JSONObject;

import android.app.Activity;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.AsyncTask;
import android.os.Bundle;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

public class TwitterProfileActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        setContentView(R.layout.twitter_profile_activity);

        String extraName = getString(R.string.tweet_extra_name);
        String tweetValue = (String)
        getIntent().getSerializableExtra(extraName);

        try {
            JSONObject tweetObject = new JSONObject(tweetValue);
            String url = tweetObject.getString("profile_image_url");
            AsyncTask<String, String, Bitmap> task = getTask();
            task.execute(url);

            TextView name = (TextView) findViewById(R.id.name);
            String username = tweetObject.getString("from_user");
            name.setText(username);

        } catch (JSONException e) {
            String imageAlert = getString(R.string.image_alert);
            Toast.makeText(getApplicationContext(), imageAlert,
            Toast.LENGTH_SHORT).show();
            e.printStackTrace();
        }

        super.onCreate(savedInstanceState);
    }

    private AsyncTask<String, String, Bitmap> getTask() {
        return new AsyncTask<String, String, Bitmap>() {

```

```

@Override
protected Bitmap doInBackground(String... params) {
    String url = params[0];
    HttpParams httpParameters = new BasicHttpParams();
    HttpGet httpRequest = new HttpGet(url);

    DefaultHttpClient httpClient = new
        DefaultHttpClient(httpParameters);

httpClient.getConnectionManager().closeExpiredConnections();
    try {
        HttpResponse response =
            httpClient.execute(httpRequest);
        HttpEntity entity = response.getEntity();
        InputStream is = null;
        try {
            BufferedHttpEntity bufHttpEntity = new
                BufferedHttpEntity(entity);
            is = bufHttpEntity.getContent();
            Bitmap bitmap = BitmapFactory.decodeStream(is);
            return bitmap;
        } finally {
            if (is != null) {
                try {
                    is.close();
                } catch (IOException e) {
                }
            }
        }
    } catch (ClientProtocolException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        httpClient.getConnectionManager().
            closeExpiredConnections();
    }
    return null;
}

@Override
protected void onPostExecute(Bitmap result) {
    ImageView image = (ImageView) findViewById(R.id.pic);
    image.setImageBitmap(result);
    super.onPostExecute(result);
}
};
}
}
}

```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="android.cohort" android:versionCode="1" android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8" />

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".TwitterNameActivity"
android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER"
/>
            </intent-filter>
        </activity>
        <activity android:name=".TwitterSearchResultsActivity"
            android:label="@string/app_name">
        </activity>
        <activity android:name=".TwitterProfileActivity"
            android:label="@string/app_name">
        </activity>
        <activity android:name=".TwitterSearchResultsWithImagesActivity"
            android:label="@string/app_name">
        </activity>

    </application>
    <uses-permission android:name="android.permission.INTERNET" />
</manifest>
```

strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">TwitterReader</string>
    <string name="search">Search</string>
    <string name="search_hint">Enter Twitter Search</string>
    <string name="image_alert">Could not read profile information</string>
    <string name="empty_string_alert">The search is empty, please enter a valid
search.</string>
    <string name="searching_for_tweets">Searching for tweets</string>
    <string name="searching_for_tweets_title">Searching</string>
    <string name="search_extra_name">SEARCH</string>
    <string name="tweet_extra_name">TWEET</string>
</resources>
```

tweet_preview_with_image.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="200dp"
    android:gravity="center_vertical">
    <ImageView android:id="@+id/tweet_user" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <TextView android:id="@+id/tweet" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>
```

tweet_preview.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="200dp"
    android:gravity="center_vertical">
    <TextView android:id="@+id/tweet_user" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <TextView android:id="@+id/tweet" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>
```

twitter_name_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:gravity="center">
    <EditText
        android:id="@+id/search_text"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:hint="@string/search_hint"/>
    <Button
        android:id="@+id/search"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/search"/>
</LinearLayout>
```

twitter_profile_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent">
    <TextView android:id="@+id/name" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <ImageView android:id="@+id/pic" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>
```

twitter_search_results_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent">
    <ListView android:id="@+id/listView" android:layout_width="fill_parent"
        android:layout_height="fill_parent" />
</LinearLayout>
```

```

package android.cohort;

import java.io.IOException;
import java.util.ArrayList;

import org.apache.http.HttpResponse;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.params.BasicHttpParams;
import org.apache.http.params.HttpParams;
import org.apache.http.util.EntityUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListAdapter;
import android.widget.ListView;
import android.widget.TextView;

public class TwitterSearchResultsActivity extends Activity {

    private ProgressDialog progressDialog;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        String extraName = getString(R.string.search_extra_name);
        String searchValue = (String)
            getIntent().getSerializableExtra(extraName);

        setContentView(R.layout.twitter_search_results_activity);

        AsyncTask<String, String, String> task = getAsyncTask();
        task.execute(searchValue);

        super.onCreate(savedInstanceState);
    }

    protected void parseResults(String jsonString) {
        try {
            JSONObject results = new JSONObject(jsonString);
            JSONArray resultsArray = results.getJSONArray("results");
            ArrayList<JSONObject> resultList = new ArrayList<JSONObject>();

            for (int i = 0; i < resultsArray.length(); i++){
                JSONObject jsonTweet = resultsArray.getJSONObject(i);
                resultList.add(jsonTweet);
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
}

```

```

        }
        setListAdapter(resultList);
    } catch (JSONException e) {
        e.printStackTrace();
    }
}

private void setListAdapter(ArrayList<JSONObject> resultList) {
    ListView x = (ListView) findViewById(R.id.listView);
    x.setAdapter(getArrayAdapter(resultList));
}

private ListAdapter getArrayAdapter(ArrayList<JSONObject> tweets) {
    return new ArrayAdapter<JSONObject>(getApplicationContext(),
        R.layout.tweet_preview, tweets) {
        public View getView(int position, View v, ViewGroup parent) {
            if (v == null) {
                LayoutInflater vi = (LayoutInflater)
                    getSystemService(Context.LAYOUT_INFLATER_SERVICE);
                v = vi.inflate(R.layout.tweet_preview, null);
            }

            JSONObject tweet = getItem(position);

            TextView userView = (TextView)
                v.findViewById(R.id.tweet_user);

            try{
                String userString = tweet.getString("from_user");
                userView.setText(userString);
            }catch (JSONException e){
                userView.setText("NO_USERNAME");
            }

            TextView tweetView = (TextView) v.findViewById(R.id.tweet);

            try{
                String tweetString = tweet.getString("text");
                tweetView.setText(tweetString);
            }catch (JSONException e){
                tweetView.setText("NO_TWEET");
            }

            v.setOnClickListener(getOnClickListener(tweet));

            return v;
        }
    };
}

protected OnClickListener getOnClickListener(JSONObject tweet) {
    final String tweetString = tweet.toString();
    return new OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent twitterProfileActivity = new
                Intent(getApplicationContext(),

```



```

        TwitterProfileActivity.class);
        String extraName = getString(R.string.tweet_extra_name);
        twitterProfileActivity.putExtra(extraName, tweetString);
        startActivity(twitterProfileActivity);
    }
};
}

public AsyncTask<String, String, String> getAsyncTask(){
    return new AsyncTask<String, String, String>() {
        @Override
        protected void onPreExecute() {
            String searchingForTweetsTitle =
                getString(R.string.searching_for_tweets_title);
            String searchingForTweets =
                getString(R.string.searching_for_tweets);
            progressDialog =
                ProgressDialog.show(TwitterSearchResultsActivity.this,
                    searchingForTweetsTitle, searchingForTweets);
        }

        @Override
        protected String doInBackground(String... params) {
            String tweets = getTweets(params[0]);
            return tweets;
        }

        private String getTweets(String string) {
            String url = "http://search.twitter.com/search.json?q=" +
                string.trim();
            HttpParams httpParameters = new BasicHttpParams();
            HttpGet httpRequest = new HttpGet(url);

            DefaultHttpClient httpClient = new
                DefaultHttpClient(httpParameters);

            httpClient.getConnectionManager().
                closeExpiredConnections();
            try {
                HttpResponse response =
                    httpClient.execute(httpRequest);
                String result =
                    EntityUtils.toString(response.getEntity());
                return result;
            } catch (ClientProtocolException e) {
                e.printStackTrace();
            } catch (IOException e) {
                e.printStackTrace();
            } finally {
                httpClient.getConnectionManager().
                    closeExpiredConnections();
            }
            return null;
        }
    }
}

```

```
        protected void onPostExecute(String result) {
            parseResults(result);
            progressDialog.dismiss();
            progressDialog = null;
        };
    };
}
```

```

package android.cohort;

import java.io.IOException;
import java.io.InputStream;
import java.util.ArrayList;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.entity.BufferedHttpEntity;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.params.BasicHttpParams;
import org.apache.http.params.HttpParams;
import org.apache.http.util.EntityUtils;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ImageView;
import android.widget.ListAdapter;
import android.widget.ListView;
import android.widget.TextView;

public class TwitterSearchResultsWithImagesActivity extends Activity {

    private ProgressDialog progressDialog;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        String extraName = getString(R.string.search_extra_name);
        String searchValue = (String)
getIntent().getSerializableExtra(extraName);

        setContentView(R.layout.twitter_search_results_activity);

        AsyncTask<String, String, String> task = getAsyncTask();
        task.execute(searchValue);

        super.onCreate(savedInstanceState);
    }

    protected void parseResults(String jsonString) {
        try {

```

```

        JSONObject results = new JSONObject(jsonString);
        JSONArray resultsArray = results.getJSONArray("results");
        ArrayList<JSONObject> resultList = new ArrayList<JSONObject>();

        for (int i = 0; i < resultsArray.length(); i++) {
            JSONObject jsonTweet = resultsArray.getJSONObject(i);
            resultList.add(jsonTweet);
        }
        setListAdapter(resultList);
    } catch (JSONException e) {
        e.printStackTrace();
    }
}

private void setListAdapter(ArrayList<JSONObject> resultList) {
    ListView x = (ListView) findViewById(R.id.listView);
    x.setAdapter(getArrayAdapter(resultList));
}

private ListAdapter getArrayAdapter(ArrayList<JSONObject> tweets) {
    return new ArrayAdapter<JSONObject>(getApplicationContext(),
R.layout.tweet_preview_with_image, tweets) {
        public View getView(int position, View v, ViewGroup parent) {
            if (v == null) {
                LayoutInflater vi = (LayoutInflater)
getSystemService(Context.LAYOUT_INFLATER_SERVICE);
                v = vi.inflate(R.layout.tweet_preview_with_image,
null);
            }

            JSONObject tweet = getItem(position);

            try {
                final String imageUrl =
tweet.getString("profile_image_url");
                final ImageView image = (ImageView)
v.findViewById(R.id.tweet_user);

                if (image.getTag() != null &&
image.getTag().equals(imageUrl))
                    return v;

                image.setImageBitmap(null);
                image.setTag(imageUrl);
                (new AsyncTask<String, String, Bitmap>() {
                    @Override
                    protected Bitmap doInBackground(String...
params) {
                        String url = params[0];
                        HttpParams httpParameters = new
BasicHttpParams();
                        HttpGet httpRequest = new HttpGet(url);
                        DefaultHttpClient httpClient = new
DefaultHttpClient(httpParameters);

```

```

        httpClient.getConnectionManager().closeExpiredConnections();
        try {
            HttpResponse response =
httpClient.execute(httpRequest);
            HttpEntity entity =
response.getEntity();
            InputStream is = null;
            try {
                BufferedHttpEntity
bufHttpEntity = new BufferedHttpEntity(entity);
                is =
bufHttpEntity.getContent();
                Bitmap bitmap =
BitmapFactory.decodeStream(is);
                return bitmap;
            } finally {
                if (is != null) {
                    try {
                        is.close();
                    } catch (IOException e)
{
                        }
                }
            }
        } catch (ClientProtocolException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            httpClient.getConnectionManager().closeExpiredConnections();
        }
        return null;
    }

    @Override
    protected void onPostExecute(Bitmap result) {

        if (!image.getTag().equals(imageUrl))
            return;

        image.setImageBitmap(result);
        super.onPostExecute(result);
    }
    }).execute(imageUrl);
} catch (JSONException e) {
}

TextView tweetView = (TextView) v.findViewById(R.id.tweet);

try {
    String tweetString = tweet.getString("text");
    tweetView.setText(tweetString);
} catch (JSONException e) {
    tweetView.setText("NO_TWEET");
}

```

```

        v.setOnClickListener(getOnClickListener(tweet));
        return v;
    }
};
}

protected OnClickListener getOnClickListener(JSONObject tweet) {
    final String tweetString = tweet.toString();
    return new OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent twitterProfileActivity = new
Intent(getApplicationContext(), TwitterProfileActivity.class);
            String extraName = getString(R.string.tweet_extra_name);
            twitterProfileActivity.putExtra(extraName, tweetString);
            startActivity(twitterProfileActivity);
        }
    };
}

public AsyncTask<String, String, String> getAsyncTask() {
    return new AsyncTask<String, String, String>() {
        @Override
        protected void onPreExecute() {
            String searchingForTweetsTitle =
getString(R.string.searching_for_tweets_title);
            String searchingForTweets =
getString(R.string.searching_for_tweets);
            progressDialog =
ProgressDialog.show(TwitterSearchResultsWithImagesActivity.this,
searchingForTweetsTitle, searchingForTweets);
        }

        @Override
        protected String doInBackground(String... params) {
            String tweets = getTweets(params[0]);
            return tweets;
        }

        private String getTweets(String string) {
            String url = "http://search.twitter.com/search.json?q=" +
string.trim();

            HttpParams httpParameters = new BasicHttpParams();
            HttpGet httpRequest = new HttpGet(url);

            DefaultHttpClient httpClient = new
DefaultHttpClient(httpParameters);

            httpClient.getConnectionManager().closeExpiredConnections();
            try {
                HttpResponse response =
httpClient.execute(httpRequest);
                String result =
EntityUtils.toString(response.getEntity());
                return result;
            } catch (ClientProtocolException e) {

```

```
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        httpClient.getConnectionManager().closeExpiredConnections();
    }
    return null;
}

protected void onPostExecute(String result) {
    parseResults(result);
    progressDialog.dismiss();
    progressDialog = null;
};
}
}
```