**Project Resources**Until the Android Basics: Networking course is live, please use the following links and tutorials to continue learning and working on the networking projects:

[Performing Network Operations](https://developer.android.com/training/basics/network-ops/index.html)

* Learn how to connect to the Internet and transfer data

[JSON Parsing Tutorial](http://www.tutorialspoint.com/android/android_json_parser.htm)

* Learn how to parse the JSON you get from internet APIs on Android

**Step 1  
  
Project Overview**

The final project is a chance for you to combine and practice everything you learned in this section of the Nanodegree program. You will be making your own app that connects to the Internet to provide news articles on a topic of your choice.

The goal is to create a News feed app which gives a user regularly-updated news from the internet related to a particular topic, person, or location. The presentation of the information as well as the topic is up to you.

**Why this project?**

In the most recent portion of the Nanodegree program, you worked to build the Earthquake app. Along the way, you learned about connecting to the internet in Android, parsing responses from an API, updating the information in your app, and properly displaying that information. Practicing these skills is imperative in order to build apps that delight and surprise users by anticipating their needs and supplying them with relevant information.

**What will I learn?**

This project is about combining various ideas and skills we’ve been practicing throughout the course. They include:

* Connecting to an API
* Parsing the response of the API
* Handling error cases gracefully
* Updating information regularly

**Subscription vs Free-Version student**

If you are subscribed to the paid Nanodegree program, coaches will be available in the Coaches Lounge to help guide you through the final project. You will receive feedback after submitting your project and a verified certificate after successfully finishing this project.

If you are enrolled in the free version of this course, the project rubric and all of the project details are available to you.

Either way, we would love to see a description and link to your app on the discussion forum.

**Step 2:**

To Learn about the Google Feed API, take a look at [this page](https://developers.google.com/feed/v1/jsondevguide#using_json). Also, try modifying [this example query](https://ajax.googleapis.com/ajax/services/feed/find?v=1.0&q=dance).

Your project will be evaluated using the [News App project rubric](https://review.udacity.com/#!/rubrics/165/view).

**Additional Criteria**

The intent of this project is to give you practice writing raw Java code using the necessary classes provided by the Android framework; therefore, the use of external libraries will not be permitted to complete this project.

**Step 3:**

**Prepare for Submission**

**Clean Your Build**

Before submitting, please follow the instructions for cleaning your project files. This removes some temporary files and greatly decreases the size of your project.

[Cleaning your project files](https://d17h27t6h515a5.cloudfront.net/topher/2016/June/5769c116_1000-files-tutorial/1000-files-tutorial.pdf)

**Review the Project Rubric**

Udacity reviewers will be reviewing your project based on the project rubric. All specifications must be met to pass the project.

### [Project review RubrRequires Changes](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

#### [3 SPECIFICATIONS REQUIRE CHANGES](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

[Hi! You have done a great job so far! There are some details to fix but you are on the good way. Keep on working like this!](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

### [Layout](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

**[App contains a ListView which becomes populated with list items.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[List items display at least author and title information.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[When there are more list items than fit on the screen, the app allows scrolling through the list.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[The code adheres to all of the following best practices:](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

* **[Text sizes are defined in sp](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**
* **[Lengths are defined in dp](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**
* **[Padding and margin is used appropriately, such that the views are not crammed up against each other.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[Information displayed on list items is not crowded.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[The content should not be cut off if the device is rotated to landscape mode.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

[Your ListView items are disappearing when the screen orientation change, you can take a look to the](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)[[Oficial Android documentation](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)](https://developer.android.com/guide/topics/resources/runtime-changes.html#HandlingTheChange)[so you can persist your ListView data by saving the state of your activity, on screen rotation.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

### [Functionality](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

**[The code runs without errors.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[The app fetches data via an HTTP request from the Google Books API, using a class such as HttpUriRequest or HttpURLConnection.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[The result of the request is validated to account for a bad server response or lack of server response.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

[You have done a great job! If there is no internet connection, you could show a Toast message to the user informing him that there is no internet connection and the search could not be completed.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

**[The network call occurs off the UI thread using an AsyncTask or similar threading object.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

[Great job!](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

**[The JSON response is parsed correctly, and relevant information is stored in the app.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

[If the search has results without authors, the JSON is not parsed correctly.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

**[The ListView is properly populated with the information parsed from the JSON response.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[When there is no data to display, the app shows a default TextView that informs the user how to populate the list.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[The intent of this project is to give you practice writing raw Java code using the necessary classes provided by the Android framework; therefore, the use of external libraries for core functionality will not be permitted to complete this project.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

### [Code Readability](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

**[All variables, methods, and resource IDs are descriptively named such that another developer reading the code can easily understand their function.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

**[The code is properly formatted i.e. there are no unnecessary blank lines; there are no unused variables or methods; there is no commented out code.  
The code also has proper indentation when defining variables and methods.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**

[RESUBMIT PROJECT](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

[**[DOWNLOAD PROJECT](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)**](https://udacity-github-sync-content.s3.amazonaws.com/_submissions/187752_gsqe4.zip)

[Learn the](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)[[best practices for revising and resubmitting your project](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)](https://review.udacity.com/)[.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

[Have a question about your review? Email us at](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)[[review-support@udacity.com](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)](mailto:review-support@udacity.com)[and include the link to this review.](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

[ic](https://review.udacity.com/?_ga=1.193267053.1362034947.1463828731" \l "!/rubrics/165/view)

**Final Submission Checklist**

Before submitting your project for evaluation, we recommend that you check that each of the following is true:

1. Your app compiles and runs as expected.
2. You are proud of your app and its output.
3. You completed this project according to instructions.
4. You cleaned the project using the instructions above.
5. You checked your project against the rubric.

**Zip your Project for Submission**

Be sure to zip your whole android project only after you have cleaned it.