## Takeaway Android Assignment

### Goal

The goal of this assignment is to implement a sample project, where you visualize a restaurant list. You are able to sort the restaurant list based on its current openings state and you can select a sort value to further sort the list. Additionally, we would like to see you add an option to filter the restaurant list based on the restaurant's name.

The list item should contain the name of the restaurants, the current opening state, the selected sorting and the sort value for a restaurant.

Sort options should also be in a logical order when they are selected. Eg. Popularity descending, distance ascending etc.

### Description

In the attachment you can find a JSON file (sample.json), this file contains all the necessary data to complete this assignment. Parse the JSON file and visualize it as a restaurant list. Use the following rules for the sorting:

1. **Restaurant open state:** The list has to constantly be sorted by the restaurant open state. Open restaurants should be on the top of the list, order-ahead restaurants in the middle, closed restaurants in the bottom. (Values available in sample.json)
2. **Sort options**: Users should be able to apply additional sorting methods and this can be best-match, newest, rating average, distance, popularity, average product price, delivery costs or the minimum cost. (Values available in sample.json)

**Mandatory concerns**

* All functionality should be implemented in a way that strives to separate the UI from background code. The preferred way to achieve this is <https://developer.android.com/topic/libraries/architecture/viewmodel.html>
* Modular architecture is a plus !
* If any of these patterns is not used, it is **mandatory** to explain the approach used with justifications- this must achieve an equivalent level of separation to the Android Architecture Components method.
* The UI must not be laggy, buggy, unstable, or poor. Neatness and clarity are absolute requirements. Actual Artwork or graphics are **NOT** considered important. **BUT** place of use of Android UI widgets should be logical within their definition.
* Unit tests should be implemented.
* Version control system should be used and README implemented.

**Bonus :**

* UI tests
* Retain the selected sort option when the app is killed.

### 

Good luck!

Takeaway.com team