JSON in Android

Mobile Application Development

JSON

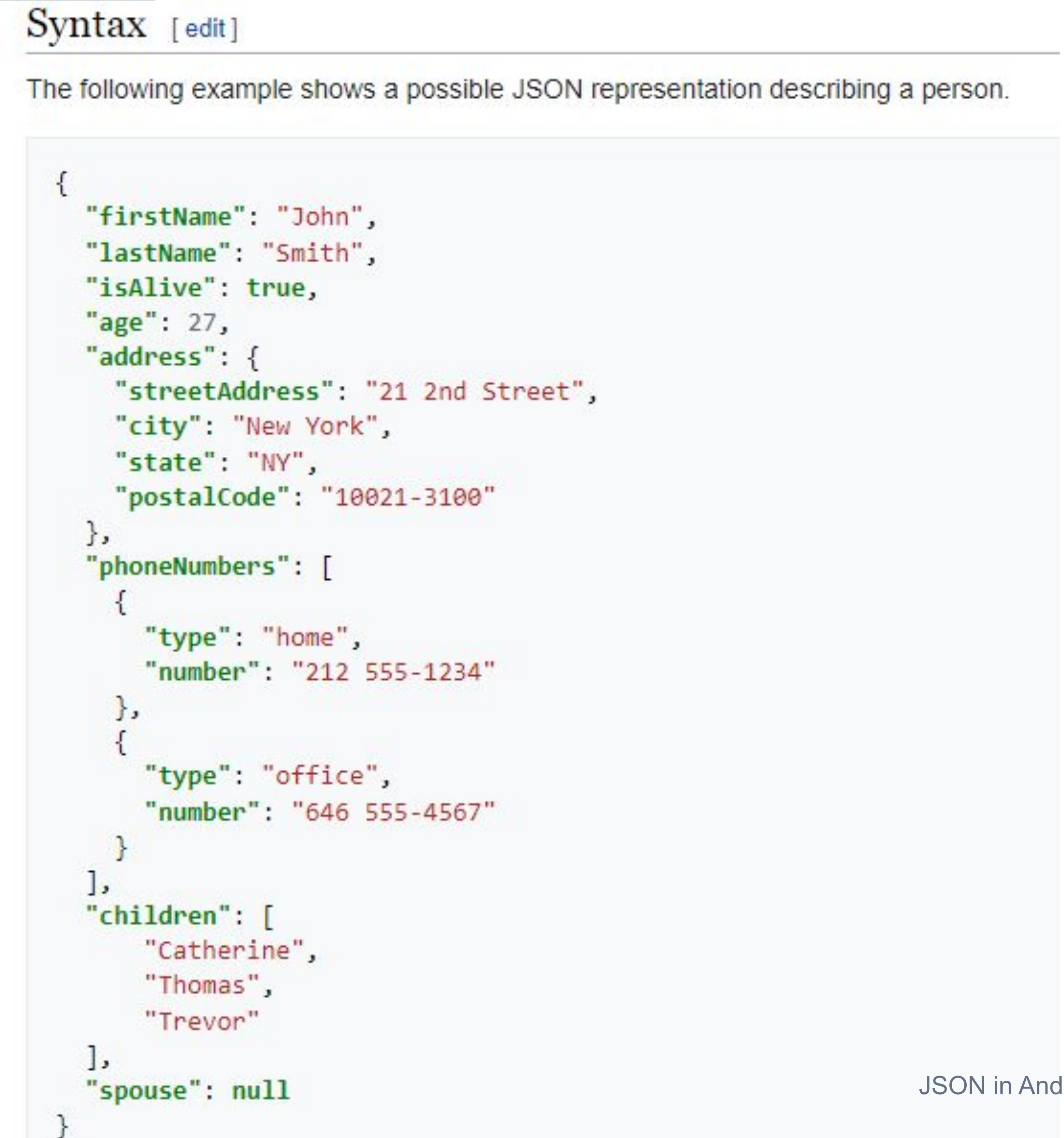
- **JSON** - JavaScript Object Notation
- one of the simplest **data exchange formats** to interchange the data from the server
- best alternative for XML.
- **JSON** is an open standard file format and data interchange format that uses human-readable text to store and transmit data objects consisting of attribute–value pairs and arrays (or other serializable values).
- It is a common data format with **diverse uses in electronic data interchange.**

https://en.wikipedia.org/wiki/JSON

JSON

- JSON is a language-independent data format.
- It was derived from JavaScript, but many modern programming languages include code to generate and parse JSON-format data.
- JSON filenames use the extension `.json`.

https://en.wikipedia.org/wiki/JSON



Syntax [edit]

The following example shows a possible JSON representation describing a person.

```
{
  "firstName": "John",
  "lastName": "Smith",
  "isAlive": true,
  "age": 27,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021-3100"
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "office",
      "number": "646 555-4567"
    }
  ],
  "children": [
      "Catherine",
      "Thomas",
      "Trevor"
  ],
  "spouse": null
}
```

- Generally, the JSON nodes will start with a **square bracket** ([) or with a **curly bracket** ({).

- the **square bracket** ([) represents the starting of a **JSONArray** node

- **curly bracket** ({) represents a **JSONObject**

- so we need to call appropriate method to get the data.

Source:-https://www.tutlane.com/tutorial/android/android-json-parsing-with-examples

JSON - Data types

JSON's basic data types are:

- **Number** (signed decimal number),**String, Boolean**
- **Array**: an ordered list of zero or more elements. Arrays use square bracket notation with comma-separated elements.
- **Object**: a collection of name–value pairs where the names (also called keys) are strings. Objects are delimited with curly brackets and use commas to separate each pair, while within each pair the colon ':' character separates the key or name from its value.
- **`null`**: an empty value.

JSON in Android

Android classes to access JSON in Android

- **JSONObject**
- **JSONArray**
- **JSONStringer**
- **JSONTokener**
- **JSONException**

to parse the JSON data to get the required information in android applications.

JSONArray

- A dense indexed sequence of values.
- Values may be any mix of
    - **`JSONObjects`**,
    - other **`JSONArrays`**,
    - Strings, Booleans, Integers, Longs, Doubles,
    - `null` .
- Values may not be
    - `NaNs`, `infinities`, or of any type not listed here.

https://developer.android.com/reference/org/json/JSONArra

https://developer.android.com/reference/org/json/JSONObject

JSONArray

Constructors:

- JSONArray() - Creates a JSONArray with no values.
- JSONArray(String json) -

Creates a new JSONArray with values from the JSON string.

Methods:

JSONArray **getJSONArray**(int index)

JSONObject **getJSONObject**(int index)

Long **getLong**(int index)

String **getString**(int index)

JSONArray **put**(int index, int value)

JSONArray **put**(boolean value)

JSONObject - class

Constructor:

- `JSONObject()` - Creates a `JSONObject` with no name/value mappings.
- `JSONObject(String json)`

Creates a new `JSONObject` with name/value mappings from the JSON string.

https://developer.android.com/reference/org/json/JSONObject

JSONObject - class

<u>Methods:</u>

int `getInt(String name)`

Returns the value mapped by `name` if it exists or throws Exception otherwise.

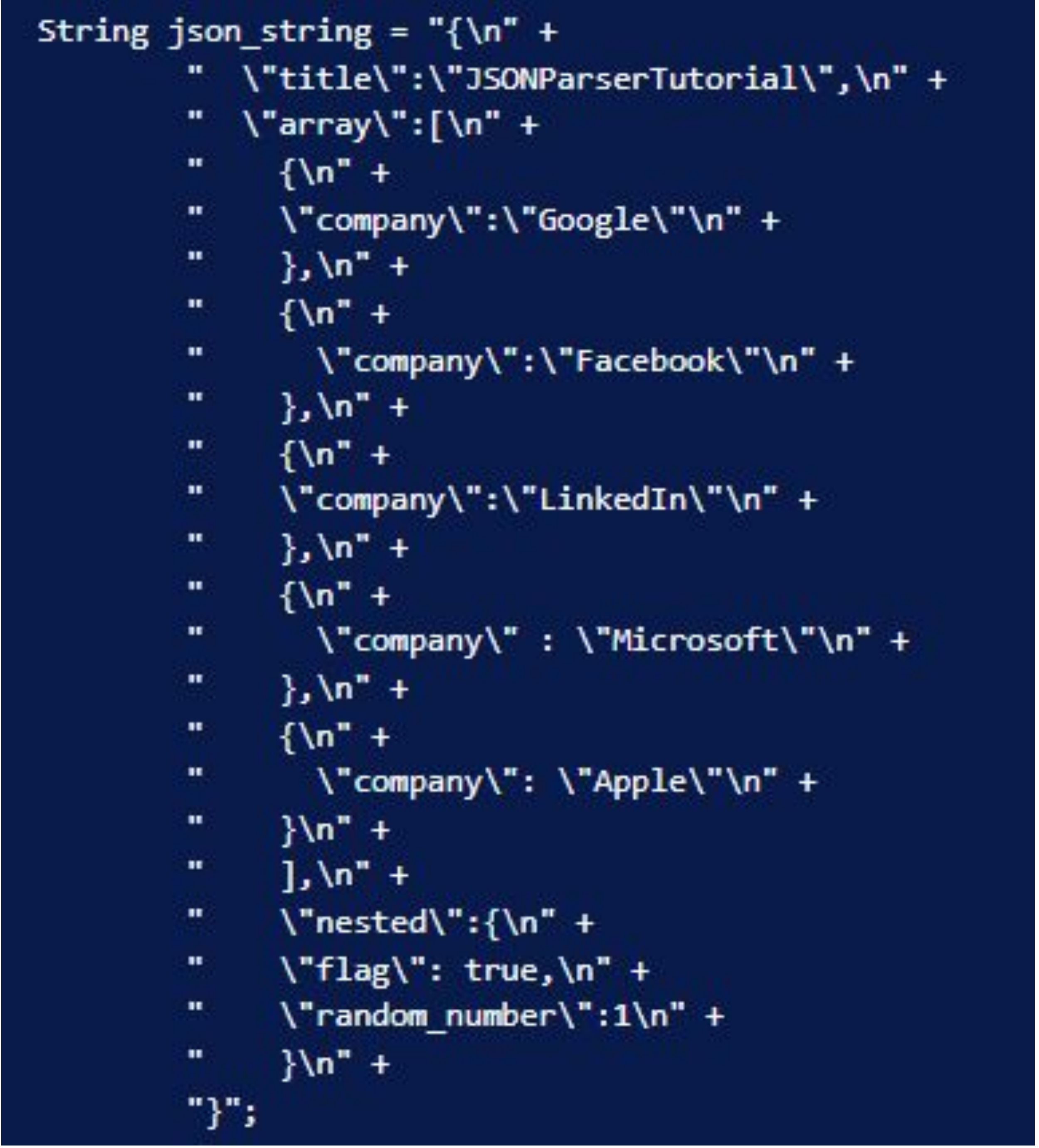
JSONArray   getJSONArray(String name)

JSONObject   getJSONObject(String name)

JSONObject   put(String name, long value)

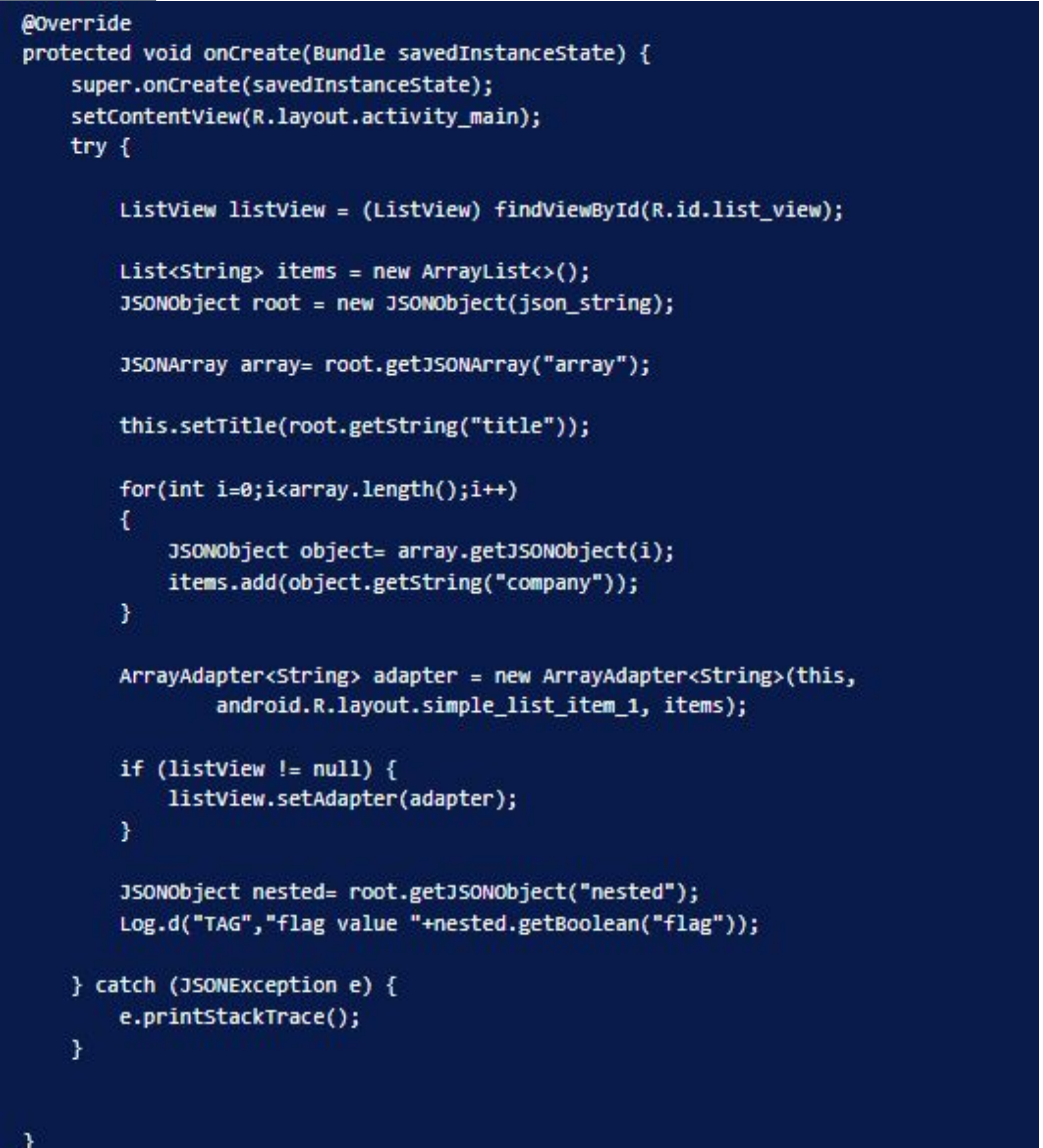
https://developer.android.com/reference/org/json/JSONObject

Example



```
String json_string = "{\n" +
        "  \"title\":\"JSONParserTutorial\",\n" +
        "  \"array\":[\n" +
        "    {\n" +
        "    \"company\":\"Google\"\n" +
        "    },\n" +
        "    {\n" +
        "      \"company\":\"Facebook\"\n" +
        "    },\n" +
        "    {\n" +
        "    \"company\":\"LinkedIn\"\n" +
        "    },\n" +
        "    {\n" +
        "      \"company\" : \"Microsoft\"\n" +
        "    },\n" +
        "    {\n" +
        "      \"company\": \"Apple\"\n" +
        "    }\n" +
        "    ],\n" +
        "    \"nested\":{\n" +
        "    \"flag\": true,\n" +
        "    \"random_number\":1\n" +
        "    }\n" +
        "}";
```

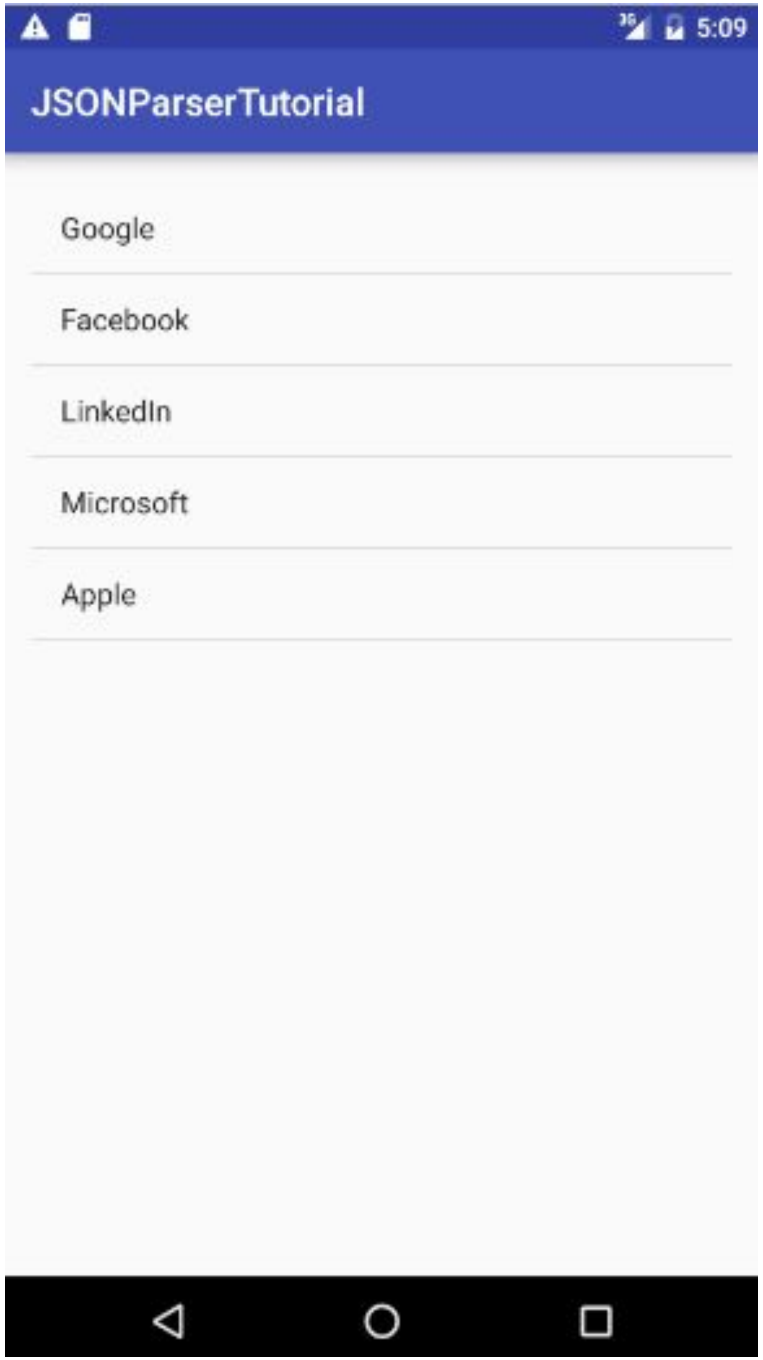
https://www.digitalocean.com/community/tutorials/android-jsonobject-json-parsing



```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    try {

        ListView listView = (ListView) findViewById(R.id.list_view);

        List<String> items = new ArrayList<>();
        JSONObject root = new JSONObject(json_string);

        JSONArray array= root.getJSONArray("array");

        this.setTitle(root.getString("title"));

        for(int i=0;i<array.length();i++)
        {
            JSONObject object= array.getJSONObject(i);
            items.add(object.getString("company"));
        }

        ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
                android.R.layout.simple_list_item_1, items);

        if (listView != null) {
            listView.setAdapter(adapter);
        }

        JSONObject nested= root.getJSONObject("nested");
        Log.d("TAG","flag value "+nested.getBoolean("flag"));

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


}
```

Example

Courtesy:
https://www.digitalocean.com/community/tutorials/android-jsonobject-json-parsing



Google

Facebook

LinkedIn

Microsoft

Apple

Example

Courtesy: https://www.digitalocean.com/community/tutorials/android-jsonobject-json-parsing