# Разбор JSON-файла при помощи библиотек
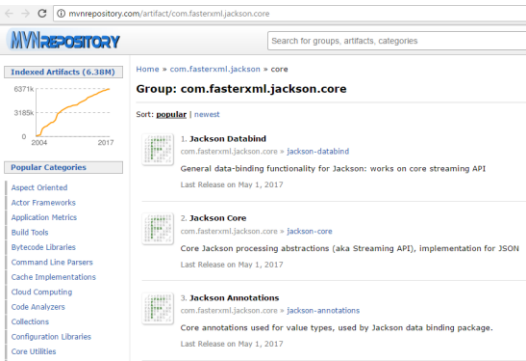
http://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-databind/2.9.0.pr3

http://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-core/2.9.0.pr3

http://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-annotations/2.9.0.pr3

| Измененная структура проекта | Содержимое файла build.gradle (Module:app) |
| --- | --- |
|  |  |

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 25
    buildToolsVersion "25.0.0"
    defaultConfig {
        applicationId "com.example.artik.androidjsontest"
        minSdkVersion 15
        targetSdkVersion 25
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
    packagingOptions {
        exclude 'META-INF/DEPENDENCIES'
        exclude 'META-INF/NOTICE'
        exclude 'META-INF/LICENSE'
        exclude 'META-INF/LICENSE.txt'
        exclude 'META-INF/NOTICE.txt'
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
        exclude group: 'com.android.support', module: 'support-annotations'
    })
    compile 'com.android.support:appcompat-v7:25.0.1'
    compile 'com.android.support.constraint:constraint-layout:1.0.2'
    // https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-databind
    compile group: 'com.fasterxml.jackson.core', name: 'jackson-databind', version: '2.9.0.pr3'
    // https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-core
    compile group: 'com.fasterxml.jackson.core', name: 'jackson-core', version: '2.9.0.pr3'
    // https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-annotations
    compile group: 'com.fasterxml.jackson.core', name: 'jackson-annotations', version: '2.9.0.pr3'

    testCompile 'junit:junit:4.12'
}
```

В файл добавлены разделы, выделенные «**жирным**»

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 25
    buildToolsVersion "25.0.0"
    defaultConfig {
        applicationId "com.example.artik.androidjsontest"
        minSdkVersion 15
        targetSdkVersion 25
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
    packagingOptions {
        exclude 'META-INF/DEPENDENCIES'
        exclude 'META-INF/NOTICE'
        exclude 'META-INF/LICENSE'
        exclude 'META-INF/LICENSE.txt'
        exclude 'META-INF/NOTICE.txt'
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
        exclude group: 'com.android.support', module: 'support-annotations'
    })
    compile 'com.android.support:appcompat-v7:25.0.1'
    compile 'com.android.support.constraint:constraint-layout:1.0.2'
    // https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-databind
    compile group: 'com.fasterxml.jackson.core', name: 'jackson-databind', version: '2.9.0.pr3'
    // https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-core
    compile group: 'com.fasterxml.jackson.core', name: 'jackson-core', version: '2.9.0.pr3'
    // https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-annotations
    compile group: 'com.fasterxml.jackson.core', name: 'jackson-annotations', version: '2.9.0.pr3'

    testCompile 'junit:junit:4.12'
}
```

## User.java

```java
package com.example.artik.androidjsontest;

public class User {
    private String name;
    int age;
    int id;
    @Override
    public String toString() { return "User{" + "name=" + name + '}'; }
    public User() {

    }
    public User(String name) { this.name = name; }
    public String getName() { return name; }
    public void setName(String name) { this.name = name; }
    public void setAge(int age) { this.age = age; }
    public int getAge() { return age; }
    public int getId() { return id; }
    public void setId(int id) {
        this.id = id;
    }
}
```

## ResponseList.java

```java
package com.example.artik.androidjsontest;
import java.util.List;
import java.util.ArrayList;

public class ResponseList {
    private List<User> users = new ArrayList<User>();

    public List<User> getUsers() {
        return users;
    }

    public void setUsers(List<User> users) {
        this.users = users;
    }
}
```

## Разбор JSON «ручной» режим

```java
if (statusCode ==  200) {
    System.out.println("Server responded with code: " + statusCode);

    InputStream stream = connection.getInputStream();
    reader = new BufferedReader(new InputStreamReader(stream));
    StringBuffer buffer = new StringBuffer();
    String line = "";

    while ((line = reader.readLine()) != null) {
        buffer.append(line);
    }
    String finalJSON = buffer.toString();
    StringBuffer finalBufferdData = new StringBuffer();
    JSONObject dataJsonObj = new JSONObject(finalJSON);
    JSONArray users = dataJsonObj.getJSONArray("users");
    for(int i = 0; i < users.length();i++)
    {
        JSONObject secondFriend = users.getJSONObject(i);
        String name = secondFriend.getString("name");
        String age = secondFriend.getString("age");
        String id = secondFriend.getString("id");
        finalBufferdData.append( "name: " + name+ "\n");
        finalBufferdData.append( "age: " + age+ "\n");
        finalBufferdData.append( "id: " + id+ "\n");
    }
    return finalBufferdData.toString(); //pases result to POstExecute
}
```

## Разбор JSON при помощи библиотек

```java
if (statusCode ==  200) {
    System.out.println("Server responded with code: " + statusCode);

    InputStream stream = connection.getInputStream();
    reader = new BufferedReader(new InputStreamReader(stream));
    StringBuffer buffer = new StringBuffer();
    String line = "";

    while ((line = reader.readLine()) != null) {
        buffer.append(line);
    }

    ResponseList responseList = new ObjectMapper().readValue(buffer.toString(), ResponseList.class);

    String finalJSON = buffer.toString();
    StringBuffer finalBufferdData = new StringBuffer();
    for(int i = 0; i < responseList.getUsers().size();i++)
    {
        String name = responseList.getUsers().get(i).getName();
        int age = responseList.getUsers().get(i).getAge();
        int id = responseList.getUsers().get(i).getId();
        finalBufferdData.append( "name: " + name+ "\n");
        finalBufferdData.append( "age: " + age+ "\n");
        finalBufferdData.append( "id: " + id+ "\n");
    }
    return finalBufferdData.toString(); //pases result to POstExecute
}
```

```java
import com.fasterxml.jackson.databind.ObjectMapper;
```