Agenda

- What is JSON?
- JSON Structure
- JSON Arrays and JSON Objects
- HTTP Calls in Android
- Main Thread vs Background Thread
- Google's GSON
- MVVM Design Pattern
- Repository Pattern
- ViewModels
- Live Data
- Observer Pattern
- Coroutines
- View Binding
- Retrofit

What is JSON?

What is JSON?

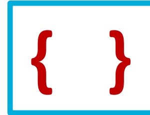
- *JSON stands for -* ***JavaScript Object Notation***
- *A very* ***lightweight*** *data-interchange format*
- *Language Independent*
- *Easy to understand*

```
{
  "movies": [
    {
      "movie": "Avengers",
      "year": 2012
    }
  ]
}
```

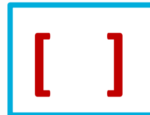
JSON Structure

- *JSON has two parts:*

  1. *JSON Objects*
     - *Contains **Key – Value** Pairs*

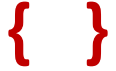


  2. *JSON Arrays*
     - *Contains JSON Objects*



JSON Objects

- *JSON Objects are represented by Curly brackets*



- *Contains KEY-VALUE pairs*

- *Key and Value is separated by colon*

- *Key-Value pairs are separated by comma*

HTTP Calls in Android

HttpUrlConnection Class

HTTP Calls

- ***HttpUrlConnection*** *Class*
- *Send and receive data over the web*
- *Data may be of any type and length*
- *Can be used to send and receive streaming data whose length is not known in advance*

Problem

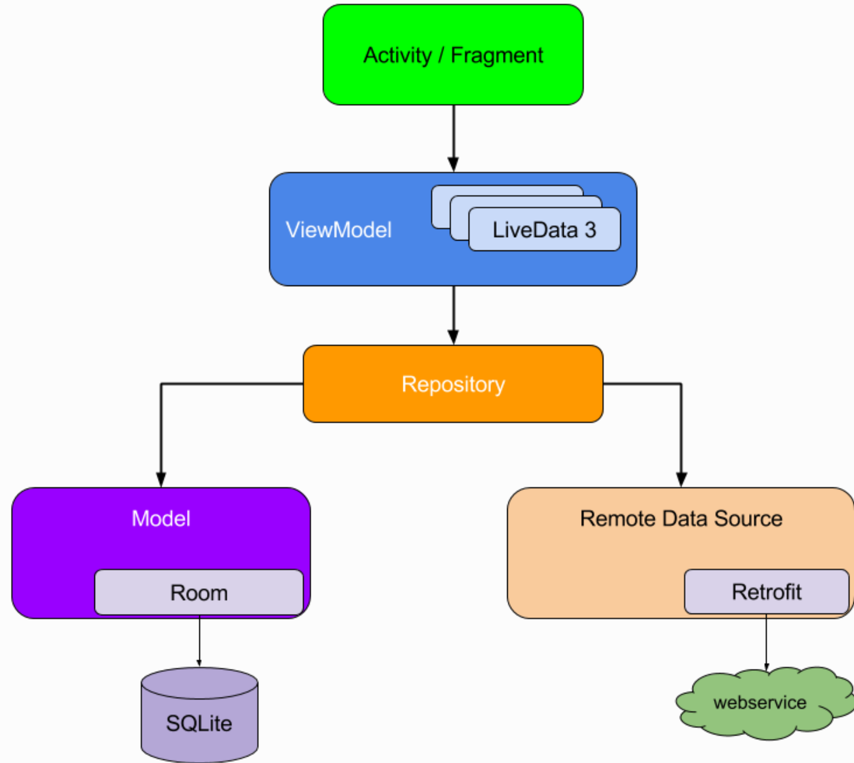
- *Android doesn’t allow to make network calls on UI thread (Main)*
- *Also, it doesn’t allow to change anything on the UI if you are on a background thread (Worker)*
- ***Coroutines** are the recommended solution for asynchronous calls.*

GSON

- *Provide simple* **toJson()** *and* **fromJson()** *methods to convert Java objects to JSON and vice-versa*

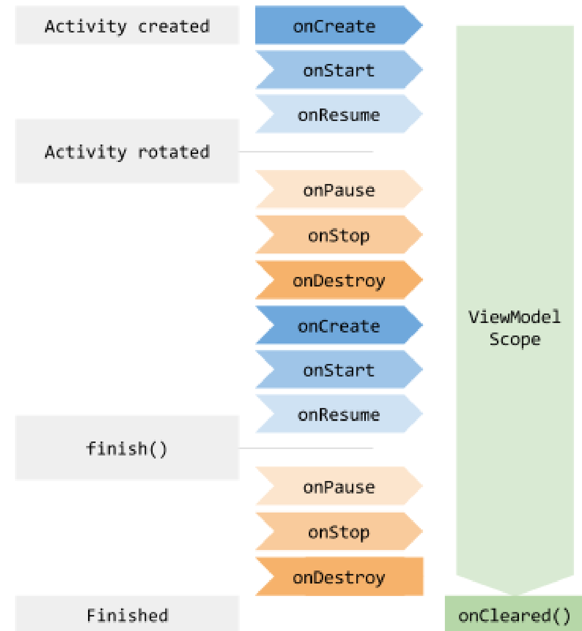
```
val movie = Gson().fromJson(jsonString, Movie::class.java)
```

MVVM



ViewModel

- *The ViewModel class is designed to store and manage UI-related data in a lifecycle conscious way.*
- *The ViewModel class allows data to survive configuration changes such as screen rotations.*

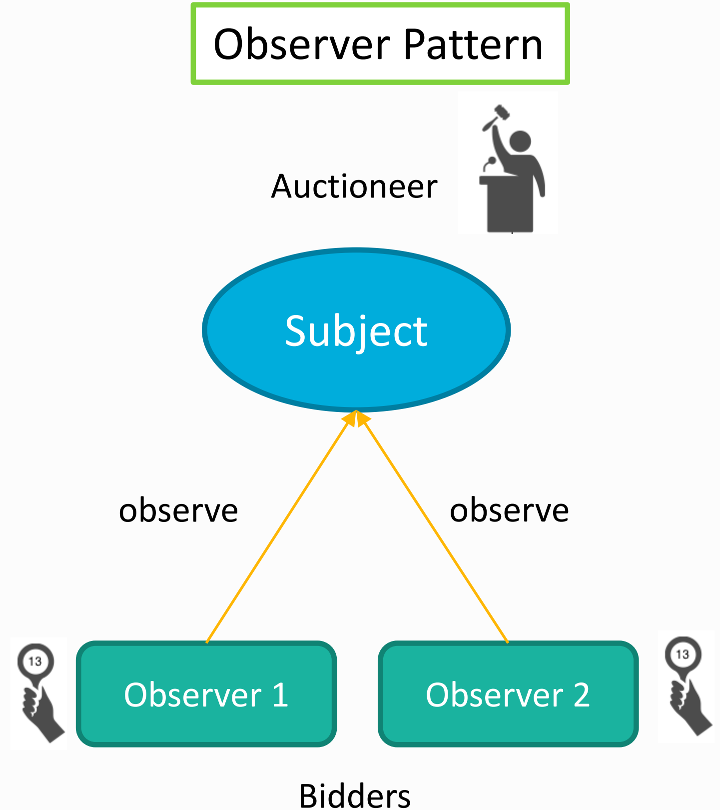


Live Data

- *is an observable data holder class*
- *is lifecycle-aware*
- *follows the observer pattern.*
- *notifies Observer objects when underlying data changes*

Observer Pattern

- *Think of Observer Pattern as an auction event*
- *Subject notifies observers directly by calling one of their methods*



Thank You….!!