# Retrofit

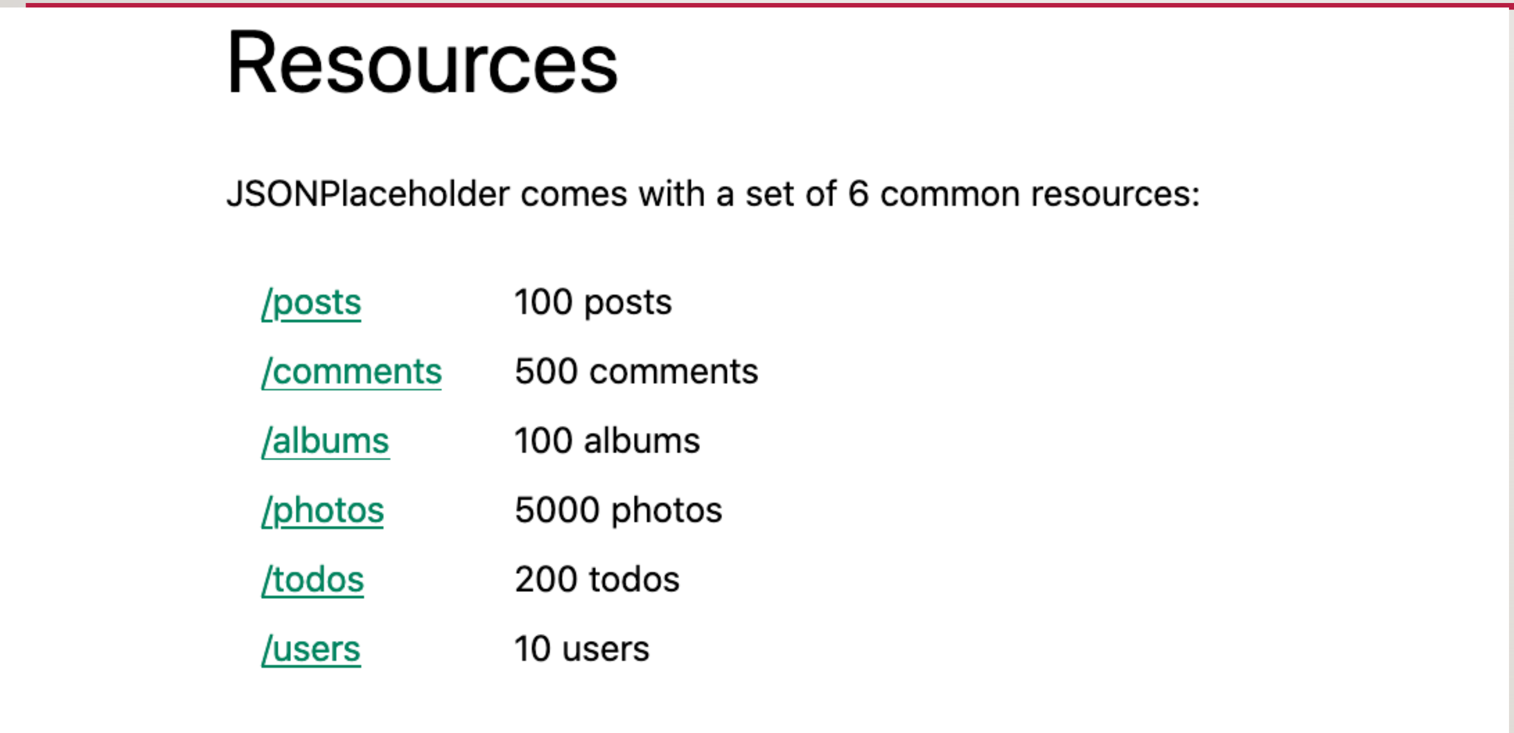
- is A well-known library
- Type-safe HTTP client for Android and Java
- Provides a complete abstraction layer , handle all underlining network operations

- Using Retrofit to retrieve data
  1. add retrofit dependencies
  2. create Interface and define methods for the Web API end points
  3. Create manager class that return implementation of the interface ( created by Retrofit)
  4. Call the methods

# Add Dependencies

- implementation 'com.squareup.retrofit2:retrofit:2.4.0’
- implementation 'com.squareup.retrofit2:converter-gson:2.4.0'

# Web Api End Points ( Resources )



## Resources

JSONPlaceholder comes with a set of 6 common resources:

| | |
|---|---|
| /posts | 100 posts |
| /comments | 500 comments |
| /albums | 100 albums |
| /photos | 5000 photos |
| /todos | 200 todos |
| /users | 10 users |

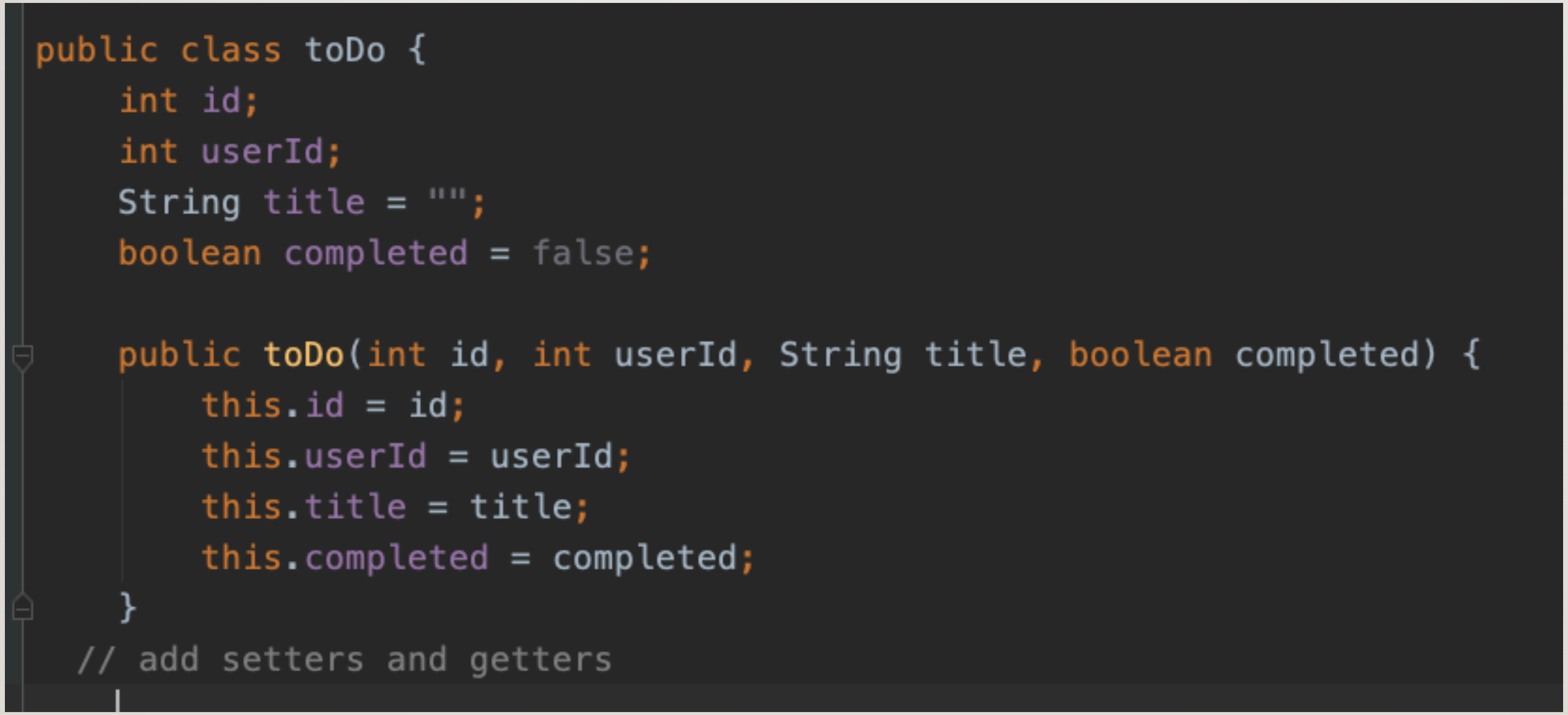
# Create Interface and define methods for the Web API end points

```
public interface ApiService {
   @GET("todos")
    Call<List<toDo>> getTodoList();
}
```

# Manager Class  to return implementation of the Interface

```
• public class ApiManager {
     static  ApiService service = null;
    static  String url ="https://jsonplaceholder.typicode.com/";
     static ApiService getService(){
         if( service == null){
        Retrofit retrofit=   new Retrofit.Builder()
                              .baseUrl(url)
                      .addConverterFactory(GsonConverterFactory.create())
                      .build();
           service = retrofit.create(ApiService.class);
         }
         return service;
     }
}
```

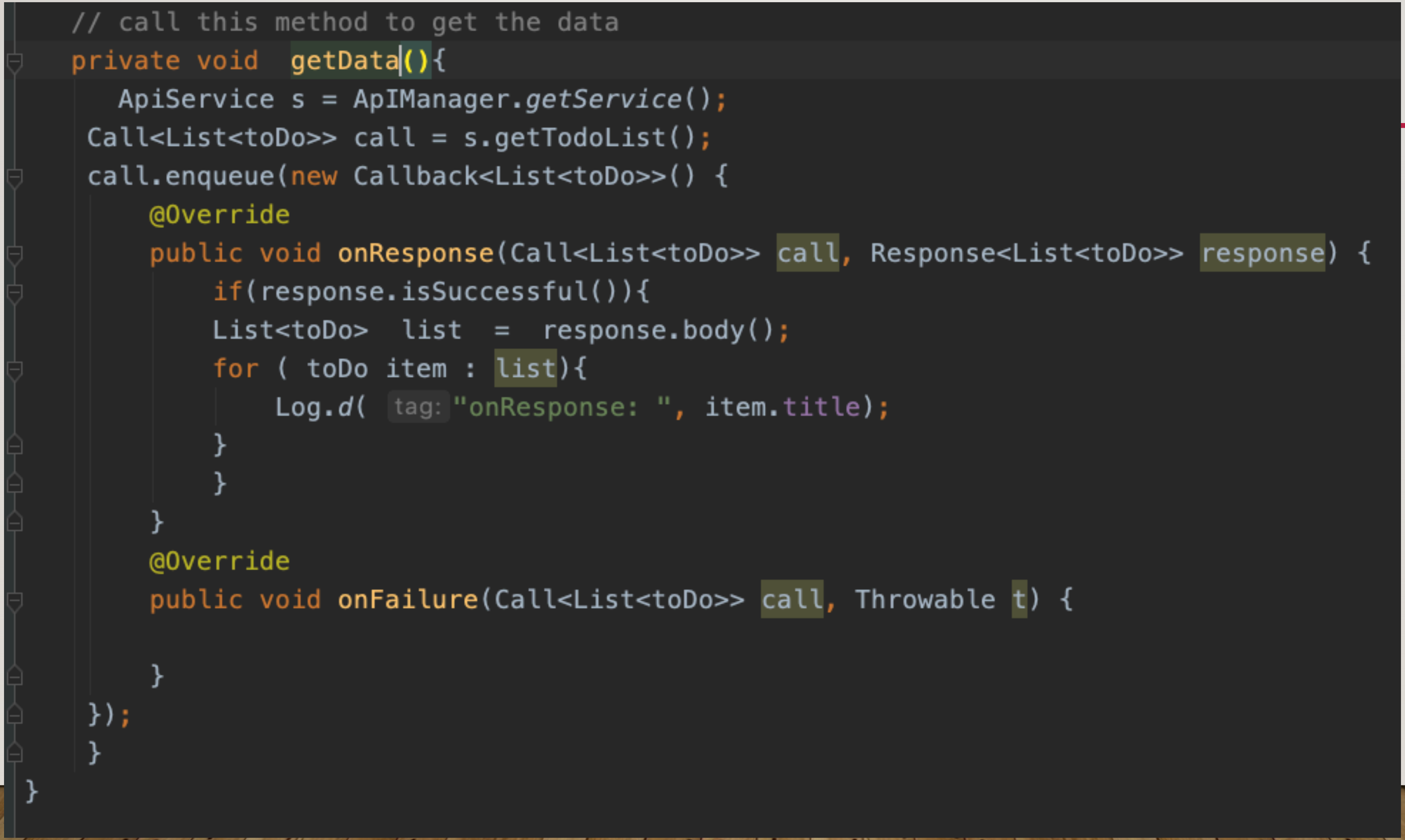
# MODEL



```java
public class toDo {
    int id;
    int userId;
    String title = "";
    boolean completed = false;

    public toDo(int id, int userId, String title, boolean completed) {
        this.id = id;
        this.userId = userId;
        this.title = title;
        this.completed = completed;
    }
  // add setters and getters
```

# Download and display data



```
   // call this method to get the data
   private void  getData(){
      ApiService s = ApIManager.getService();
    Call<List<toDo>> call = s.getTodoList();
    call.enqueue(new Callback<List<toDo>>() {
        @Override
        public void onResponse(Call<List<toDo>> call, Response<List<toDo>> response) {
            if(response.isSuccessful()){
            List<toDo>  list  =  response.body();
            for ( toDo item : list){
                Log.d( tag: "onResponse: ", item.title);
            }
            }
        }
        @Override
        public void onFailure(Call<List<toDo>> call, Throwable t) {

        }
    });
    }
}
```

# Example 1

- Get data from https://jsonplaceholder.typicode.com/todos
- Using retrofit

# Example 2

- Get list of users from
- https://jsonplaceholder.typicode.com/users
- Using retrofit

# Reference

- https://square.github.io/retrofit/