




Best Practices When Using Retrofit 🚀



Section Overview

- Monitoring network connectivity 
- Best practices when using Retrofit



Best Practices using Retrofit

- Create a single instance of Retrofit
- Create your API interface in a different file or class
- Make API network request in the background thread
- Handle different errors correctly ❌



```
@Provides
@Singleton
fun provideRetrofit(): Retrofit {
    return Retrofit.Builder()
        .baseUrl("https://imdb-api.com/en/API/")
        .build()
}
```



```
interface NewsApiService {

    @GET("api/latestNews")

    @Headers("Authorization: djehdjedhe178")
    fun latestNews(): List<News>



}
```



```
override fun getMovieDetails(id: String) = flow {
    val response = imdbApi.getMovieDetails(id = id)
}.flowOn(Dispatchers.IO)
```



Section Summary

- Monitoring network connectivity 
- Create a single instance of Retrofit
- Create your API interface in a different file or class
- Make API network request in the background thread
- Handle different errors correctly 





COURSE WRAP UP & CODE CHALLENGE

Up Next

