

Jetpack 和 Kotlin, 是时候重启追新了



扔物线 (朱凯), Android GDE, HenCoder
@rengwuxian



我为什么专注基础技术分享



我为什么又来宣传新技术



我的呼吁

这次我不想做传播核心，而更希望只作为一个传播的起点，让我们一起来把它们推开



为什么不做核心



为什么不做核心

- 什么适合单人输出？



为什么不做核心

- 什么适合单人输出?
大家都想要的东西



为什么不做核心

- 什么适合单人输出?
大家都想要的东西
- Jetpack 和 Kotlin 的推广为什么不适合单人输出?



为什么不做核心

- 什么适合单人输出?
大家都想要的东西
- Jetpack 和 Kotlin 的推广为什么不适合单人输出?
相比「学不会」，「推不动」才是真正的难点



为什么不做核心

- 什么适合单人输出?
大家都想要的东西
- Jetpack 和 Kotlin 的推广为什么不适合单人输出?
相比「学不会」，「推不动」才是真正的难点
- Jetpack 和 Kotlin 的特点：



为什么不做核心

- 什么适合单人输出？
大家都想要的东西
- Jetpack 和 Kotlin 的推广为什么不适合单人输出？
相比「学不会」，「推不动」才是真正的难点
- Jetpack 和 Kotlin 的特点：
 - 让程序写起来更快、更简单、更轻松
 - 有上手成本
 - 并没有完成现有工具做不到的事情



Jetpack 和 Kotlin 好在哪



Jetpack 和 Kotlin 好在哪

- Jetpack



Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合



Room:

```
@Entity
public class User {
    @PrimaryKey
    public int uid;

    @ColumnInfo(name = "first_name")
    public String firstName;

    @ColumnInfo(name = "last_name")
    public String lastName;
}

@Database(entities = {User.class}, version = 1)
public abstract class AppDatabase extends RoomDatabase {
    public abstract UserDao userDao();
}

AppDatabase db = Room.databaseBuilder(getApplicationContext(),
    AppDatabase.class, "database-name").build();

@Dao
public interface UserDao {
    @Query("SELECT * FROM user")
    List<User> getAll();

    @Query("SELECT * FROM user WHERE uid IN (:userIds)")
    List<User> loadAllByIds(int[] userIds);

    @Query("SELECT * FROM user WHERE first_name LIKE :first
AND "
        + "last_name LIKE :last LIMIT 1")
    User findByName(String first, String last);

    @Insert
    void insertAll(User... users);

    @Delete
    void delete(User user);
}
```

LiveData:

```
public class NameActivity extends AppCompatActivity {

    private NameViewModel mModel;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Other code to setup the activity...

        // Get the ViewModel.
        mModel = ViewModelProviders.of(this).get(NameViewModel.class);

        // Create the observer which updates the UI.
        final Observer<String> nameObserver = new Observer<String>() {
            @Override
            public void onChanged(@Nullable final String newName) {
                // Update the UI, in this case, a TextView.
                mNameTextView.setText(newName);
            }
        };

        // Observe the LiveData, passing in this activity as the LifecycleOwner and the observer.
        mModel.getCurrentName().observe(this, nameObserver);
    }
}
```


Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合



Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合
 - Jetpack 和 AndroidX



Foundation

Foundation components provide cross-cutting functionality like backwards compatibility, testing and Kotlin language support.

AppCompat

Degrade gracefully on older versions of Android

Android KTX

Write more concise, idiomatic Kotlin code

Multidex

Provide support for apps with multiple DEX files

Test

An Android testing framework for unit and runtime UI tests

Architecture

[Architecture components](#) help you design robust, testable and maintainable apps.

Data Binding

Declaratively bind observable data to UI elements

Lifecycles

Manage your activity and fragment lifecycles

LiveData

Notify views when underlying database changes

Navigation

Handle everything needed for in-app navigation

Paging

Gradually load information on demand from your data source

Room

Fluent SQLite database access

ViewModel

Manage UI-related data in a lifecycle-conscious way

WorkManager

Manage your Android background jobs

Behavior

Behavior components help your app integrate with standard Android services like notifications, permissions, sharing and the Assistant.

Download manager

Schedule and manage large downloads

Media & playback

Backwards compatible APIs for media playback and routing (including Google Cast)

Notifications

Provides a backwards-compatible notification API with support for Wear and Auto

Permissions

Compatibility APIs for checking and requesting app permissions

Sharing

Provides a share action suitable for an app's action bar

Slices

Create flexible UI elements that can display app data outside the app

Preferences

Create interactive settings screens

UI

UI components provide widgets and helpers to make your app not only easy, but delightful to use.

Animation & transitions

Move widgets and transition between screens

Auto

Components to help develop apps for Android Auto

Emoji

Enable an up-to-date emoji font on older platforms

Fragment

A basic unit of composable UI

Layout

Lay out widgets using different algorithms

Palette

Pull useful information out of color palettes

TV

Components to help develop apps for Android TV

Wear OS by Google

Components to help develop apps for Wear



Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合
 - Jetpack 和 AndroidX



Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合
 - Jetpack 和 AndroidX
- Kotlin



Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合
 - Jetpack 和 AndroidX
- Kotlin
 - 比 Java 更好用的语言



KOTLIN

```
class MainActivity : AppCompatActivity() {
```

Nullable and NonNull
types help reduce
NullPointerExceptions

```
    override fun onCreate(savedInstanceState: Bundle?) {
```

```
        ...
```

Use lambdas for concise
event handling code

```
        fab.setOnClickListener { view ->
```

```
            Snackbar.make(view, "Hello $name", Snackbar.LENGTH_LONG).show()
```

```
        }
```

```
    }
```

```
}
```

Use template expressions
in strings to avoid concatenation

Semicolons are optional

Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合
 - Jetpack 和 AndroidX
- Kotlin
 - 比 Java 更好用的语言



Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合
 - Jetpack 和 AndroidX
- Kotlin
 - 比 Java 更好用的语言
 - 更重要的：Google 的态度



Jetpack 和 Kotlin 好在哪

- Jetpack
 - Android 官方最佳实践标准的工具集合
 - Jetpack 和 AndroidX
- Kotlin
 - 比 Java 更好用的语言
 - 更重要的：Google 的态度
- 技术优势之外对于个人的价值



我们可以做什么？



我们可以做什么？

创建社区



我们可以做什么？

创建社区——我已经创建好了



什么样的社区



什么样的社区

- 社区在哪？



什么样的社区

- 社区在哪？



什么样的社区

- 社区在哪？
- 社区里有谁？



什么样的社区

- 社区在哪？
- 社区里有谁？
- 还需要谁？



什么样的社区

- 社区在哪？
- 社区里有谁？
- 还需要谁？
- 我们对未来的期望



什么样的社区

- 社区在哪?
- 社区里有谁?
- 还需要谁?
- 我们对未来的期望
- 所以，扫码加入我们吧!



DevFest

问题?



扔物线 (朱凯), Android GDE, HenCoder
@rengwuxian

