# Android 多语言

# 多语言功能支持

Android多语言功能请查看README.md，里面有完整的处理流程。

# 添加阿拉伯语支持

## 添加阿拉伯语支持

在 AndroidManifest.xml 文件中 application 节点添加支持从右到左布局方式代码

<application  
 ...  
 android:supportsRtl="true" >  
 ...  
</application>

## 添加多语言支持

具体步骤查看README.md

## 属性

android:layoutDirection： 设置组件的布局排列方向

android:textDirection： 设置组件的文字排列方向

android:textAlignment： 设置文字的对齐方式

getLayoutDirectionFromLocale()： 获取指定地区的惯用布局方式

## TextView定义文本的方向

代码定义：使用setTextDirection()方法

xml定义：android:textDirection

View.TEXT\_DIRECTION\_INHERIT(xml:inherit)默认；

View.TEXT\_DIRECTION\_FIRST\_STRONG(xml:firstStrong)第一个强制方向字符决定段落方向。 如果没有强制方向字符，则段落方向是视图的已解析布局方向；

View.TEXT\_DIRECTION\_ANY\_RTL(xml:anyRtl)如果它包含任何强制为RTL字符则段落方向是RTL，如果它包含任何强制为LTR字符则段落方向是LTR。 如果两者都没有，则段落方向是视图已解析布局方向；

View.TEXT\_DIRECTION\_LTR(xml:ltr)段落方向从左到右；

View.TEXT\_DIRECTION\_RTL(xml:rtl)段落方向从右到左；

View.TEXT\_DIRECTION\_LOCALE(xml:locale)系统判断；

View.TEXT\_DIRECTION\_FIRST\_STRONG\_LTR(xml:firstStrongLtr)第一个强制方向字符决定段落方向。 如果没有强制方向性，则段落方向为LTR；

View.TEXT\_DIRECTION\_FIRST\_STRONG\_RTL(xml:firstStrongRtl)第一个强制方向字符决定段落方向。 如果没有强制方向性，则段落方向为LTR；

android:textDirection 一般定义成跟随系统语言

android:textDirection="locale"

android:textAlignment需要和android:gravity一起使用，定义成下面形式：

android:textAlignment="viewStart"

android:gravity="start"

## 定义布局绘图的方向

android:layoutDirection

void setLayoutDirection(int);

如从右到左的布局使用在诸如阿拉伯语、希伯来语等环境中。 此属性的默认值为inherit。

View.LAYOUT\_DIRECTION\_LTR(xml:ltr)从左到右，“en-US”中使用的方向；

View.LAYOUT\_DIRECTION\_RTL(xml:rtl)从右到左；

View.LAYOUT\_DIRECTION\_INHERIT(xml:inherit)从父控件继承；

View.LAYOUT\_DIRECTION\_LOCALE(xml:locale)使用“en-US”，即从左到右；

## 文本的对齐方式

android:textAlignment

void setTextAlignment(int);

View.TEXT\_ALIGNMENT\_INHERIT(xml:inherit)默认；

View.TEXT\_ALIGNMENT\_GRAVITY(xml:gravity)相对于每个段落的文本方向；

View.TEXT\_ALIGNMENT\_TEXT\_START(xml:textStart)对其段落开头；

View.TEXT\_ALIGNMENT\_TEXT\_END(xml:textEnd)对其段落末尾；

View.TEXT\_ALIGNMENT\_CENTER(xml:center)段落居中；

View.TEXT\_ALIGNMENT\_VIEW\_START(xml:viewStart)对其视图开头，相对于layoutDirection；

View.TEXT\_ALIGNMENT\_VIEW\_END(xml:viewEnd)对其视图末尾，相对于layoutDirection；

## 数字的处理

虽然我们一般用阿拉伯数字，但是也有区分：

西阿拉伯数字：(0, 1, 2, 3, 4, 5, 6, 7, 8, 9)

东阿拉伯数字：(٠ - ١ - ٢ - ٣ - ٤ - ٥ - ٦ - ٧ - ٨ - ٩)

波斯数字：(۰ - ۱ - ۲ - ۳ - ۴ - ۵ - ۶ - ۷ - ۸ - ۹)

一般常用西阿拉伯数字，但是语言改为阿拉伯语后，数字往往显示成东阿拉伯数字形式，为了避免采用东阿拉伯数字或者波斯数字，格式化字符串与数字时可以增加Locale参数：

//String.format格式化:  
long seconds = System.*currentTimeMillis*()/1000;  
long hour = seconds/3600;  
long minutes= (seconds - hour\*3600)/60;  
String timeStr = String.format(Locale.US,"%02d:%02d:%02d",hour, minutes, seconds)  
  
//格式化数字  
long fileSize = 1024\*1024;  
DecimalFormatSymbols symbols = new DecimalFormatSymbols(Locale.US);  
DecimalFormat df = new DecimalFormat("#.00",symbols);  
  
String fileSizeString = df.format((double) fileSize / 1048576) + "MB";

网上说可以修改系统达到目的，参考：

https://blog.csdn.net/u010083774/article/details/44227929

## 阿拉伯语string.xml中的占位符

阿拉伯与常规是有些区别的，举两个例子：

string.xml(zh-rCN):

<string name="air\_quality">空气质量：%1$s</string>  
<string name="air\_index">空气指数：%1$d</string>

string.xml(ar):

<string name="air\_quality">s$1%:جودة الهواء</string>  
<string name="air\_index">d$1%:مؤشر الهواء</string>

# 适配细节

## TextView

在styles统一风格，TextView统一引用：

styles.xml:

<resources>  
 <style name="text\_rtl\_style">  
 <item name="android:textAlignment">viewStart</item>  
 <item name="android:textDirection">locale</item>  
 <item name="android:gravity">start</item>  
 </style>  
</resources>

xml:

<TextView  
 android:id="@+id/tv\_test1"

style="@style/text\_rtl\_style"

…

/>

注意，如果TextView本来就有设置anroid:gravity=“center”的，则不需要使用这个style。

# 参考文档

Android 中东阿拉伯语适配

<https://juejin.cn/post/6844903491710746631>

Android 中东阿拉伯语适配，看这一篇够了

<https://juejin.cn/post/6844903491341647879>

ANDROID阿拉伯适配RTL

<https://www.freesion.com/article/5856257571/>

Android国际化，阿语RTL适配总结

https://blog.csdn.net/jw\_66666/article/details/124035118

Android 本地化适配：RTL（right-to-left） 适配清单

https://www.cnblogs.com/plokmju/p/android\_rtl.html

阿拉伯RTL适配

<https://blog.csdn.net/qq_24535745/article/details/125064883>

Android 国际化与本地化探索

<https://blog.csdn.net/u010976213/article/details/115012368>

式样

微软雅黑 Light 5号

段落 固定值 23磅

# 目录1

## 目录2

### 目录3

#### 目录4

##### 目录5