



# آموزش برنامه نویسی اندروید در محیط اندروید استودیو

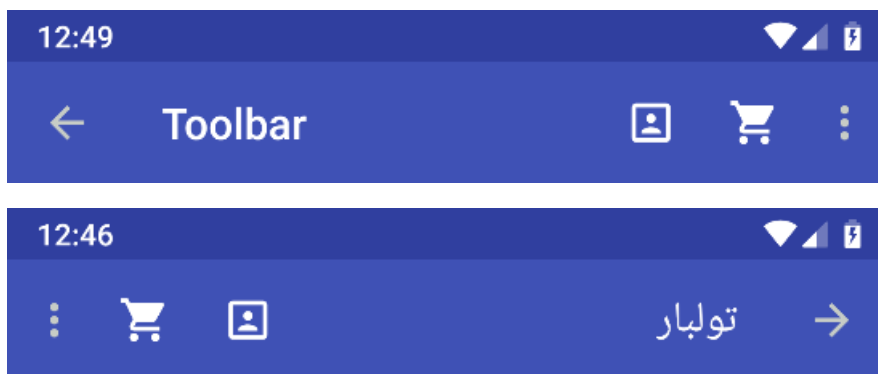
راست چین کردن Toolbar

مدرس : سیدمهدی مطهری

[www.android-studio.ir](http://www.android-studio.ir)



## به نام خدا



در مبحث آموزش ساخت **Toolbar** با نحوه پیاده سازی تولبار در پروژه اندرویدی آشنا شدیم. در این قسمت از آموزش برنامه نویسی اندروید نحوه‌ی راست چین کردن **Toolbar** در اندروید را بررسی می‌کنیم.

## ضرورت راست چین کردن اجزای رابط کاربری اندروید

مخاطبان برنامه‌های اندرویدی که شما به عنوان یک توسعه دهنده منتشر می‌کنید عمدتاً فارسی زبان هستند بنابراین لازم است چینش محتوا و قسمت‌های مختلف اپلیکیشن برای یک کاربر فارسی زبان بهینه سازی شده باشد.

مفاهیمی مانند راست چین (Right to left یا به اختصار RTL) و چپ چین (Left to right یا LTR) محدود به اپلیکیشن‌های موبایلی نبوده و حتماً در سایر محیط‌های نرم افزاری با این اصطلاحات آشنا شده‌اید. به عنوان مثال در یک ویراستار متن مانند Office Word یا صفحات وب نیز این مفاهیم به کار گرفته شده‌اند. اگر با طراحی صفحات وب آشنایی داشته باشید احتمالاً می‌دانید در CSS با تعریف `direction:rtl` تمامی محتوای موجود در صفحه راست چین می‌شود.



## راست چین کردن تولبار در اندروید

در این آموزش به بررسی روش‌های راست چین کردن Toolbar در اندروید می‌پردازیم. برای این مبحث پروژه‌ی جدیدی نمی‌سازم و همان پروژه **آموزش ساخت Toolbar** را ویرایش می‌کنم تا نتیجه کار نیز برای شما بهتر قابل درک باشد.

اندروید در API 17 یا همان Android 4.2 (Jelly Bean) و همچنین API 19 یا Android 4.4 (KITKAT) قابلیت پشتیبانی از Layout های راست به چپ (RTL Layout) را به سیستم عامل خود اضافه کرد. با معرفی این قابلیت، راه برای توسعه دهندگان اندروید تا حدود زیادی هموار شد. تا پیش از این برای راست چین کردن تولبار لازم بود تولبار کاملاً به صورت دستی پیاده سازی شود. اما از API 17 به سادگی می‌توان راست چین کردن تولبار و View های اندروید را انجام داد.

راست چین کردن Toolbar در اندروید را به دو صورت آموزش می‌دهم. روش نخست با استفاده از خاصیت `layoutDirection` در XML و روش دوم استفاده از متد `setLayoutDirection` در Java.

### ۱: راست چین کردن Toolbar توسط خاصیت `layoutDirection` در XML

این روش برای دیوایس‌های با API 17 و به بالا سازگاری دارد. قبل از هرچیز باید مطمئن شویم پروژه اندروید ما از قابلیت RTL پشتیبانی می‌کند. فعال شدن پشتیبانی از زبان‌های راست به چپ در اندروید توسط خاصیت `android:supportsRtl` در فایل `AndroidManifest.xml` صورت می‌گیرد. فایل

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="ir.android_studio.toolbar">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

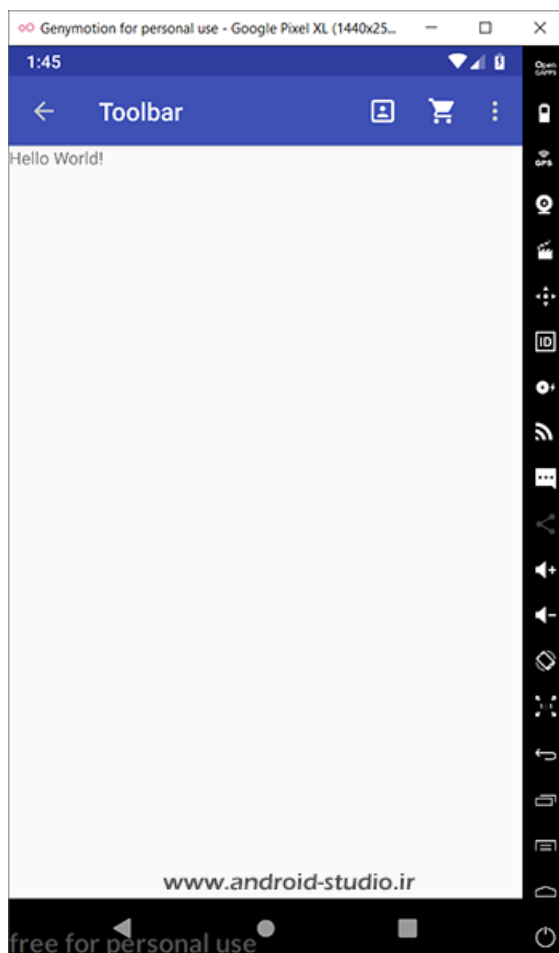
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```



مشاهده می‌کنید این خاصیت در تگ application با مقدار true قرار دارد بنابراین درون مانیفست لازم نیست کار دیگری انجام دهیم. اگر این تگ در مانیفست شما وجود ندارد آنرا اضافه کرده یا در صورتی که مقدار آن false است به true تغییر دهید.

پروژه را اجرا می‌کنم:



هنوز تغییری در تولبار صورت نگرفته و چینش در حالت LTR است.

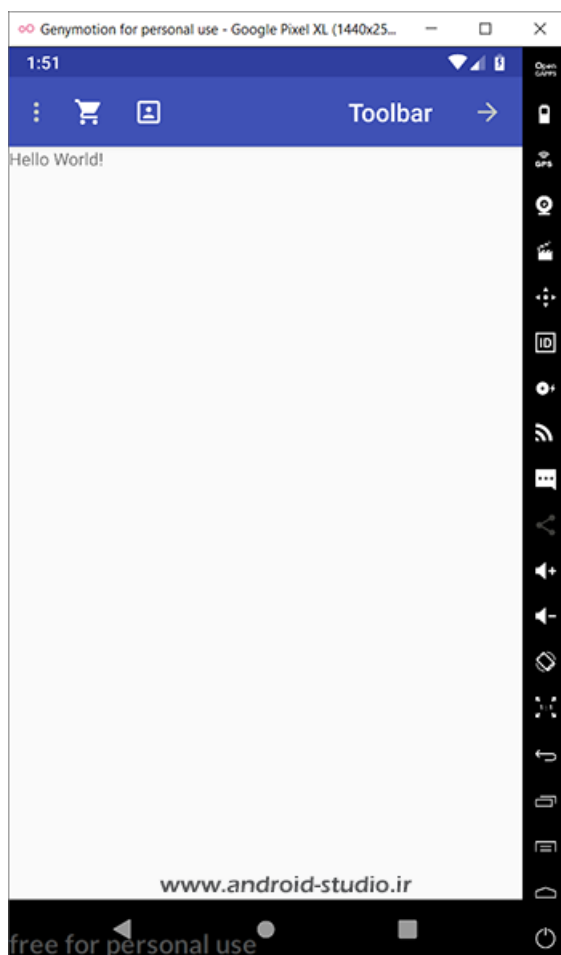
**نکته:** در صورتی که زبان دیوایس اندرویدی کاربر یکی از زبان‌های راست به چپ مانند فارسی باشد، صرف وجود `android:supportsRtl="true"` در مانیفست باعث تغییر چینش اجزاء به صورت راست چین می‌شود. با اینحال عده زیادی از کاربران فارسی زبان (از جمله خودم) بنا به دلایلی با زبان انگلیسی در محیط سیستم عامل راحتتر هستیم. بنابراین در صورتی که به خاصیت فوق اکتفا کنیم بازهم بخش عمده ای از مخاطبان فارسی زبان، اپلیکیشن ما را به صورت غیر استاندارد مشاهده خواهند کرد و لازم است مراحل بعد را هم دنبال کنیم.



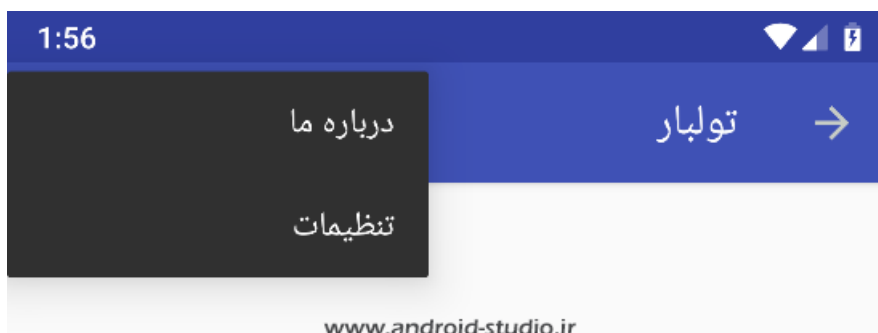
فایل activity\_main.xml را باز کرده و خاصیت layoutDirection را با مقدار rtl به Toolbar اضافه می‌کنم:

```
<android.support.v7.widget.Toolbar
    android:layout_width="match_parent"
    android:layout_height="?attr/actionBarSize"
    android:id="@+id/m_toolbar"
    android:background="?attr/colorPrimary"
    android:elevation="3dp"
    android:theme="@style/ToolbarTheme"
    app:popupTheme="@style/ThemeOverlay.AppCompat.Dark"
    android:layoutDirection="rtl"/>
```

با اجرای مجدد پروژه تولبار راست چین می‌شود:

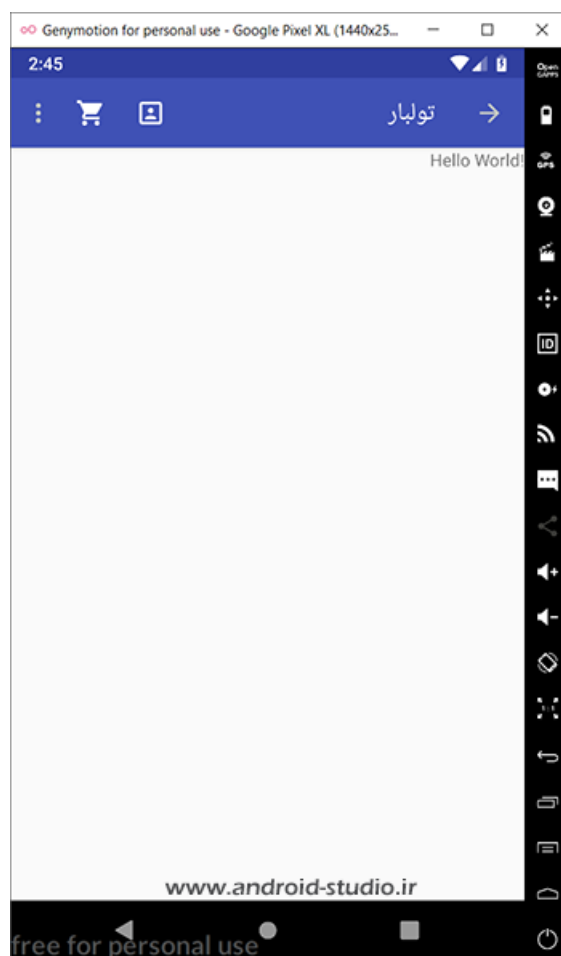


آیتم‌هایی که در سمت چپ تولبار قرار داشت به سمت راست منتقل شدند و بالعکس. حتی آیتم‌های منوی تولبار نیز راست چین شده است:



**نکته:** عنوان پروژه و آیتم‌های منو را با کلمات فارسی جایگزین کردم تا خروجی کار برای شما واضح باشد.

**نکته:** layoutDirection محدود به یک View، Widget یا Layout خاص نیست و در همه جا می‌توان استفاده کرد. در همین اکتیویتی می‌توانستم بجای اضافه کردن این خاصیت به Toolbar آنرا به لایه اصلی اکتیویتی که از نوع LinearLayout هست اضافه کنم. به این ترتیب تمامی عناصر موجود درون این لایه راست چین می‌شوند. از جمله TextView ای که عبارت Hello World را نمایش می‌دهد:





## ۲: راست چین کردن تولبار توسط متد `setLayoutDirection()`

این روش برای دیوایس‌های با API 19 و به بالا سازگاری دارد. در این روش نیز مانند قبل ابتدا باید خاصیت `android:supportsRtl` با مقدار `true` در مانیفست **پروژه اندروید** تعریف شده باشد.

خاصیت `android:layoutDirection` که در مرحله قبل به تولبار یا لایه‌ی اصلی اکتیویتی اضافه شده را حذف می‌کنم.

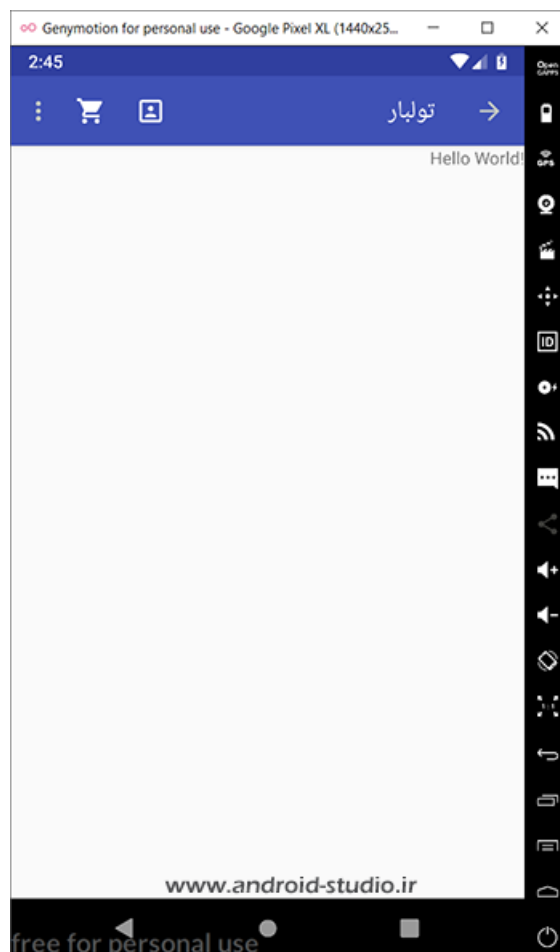
سپس خط کد زیر را داخل متد `onCreate` اکتیویتی اضافه می‌کنم:

```
getWindow().getDecorView().setLayoutDirection(View.LAYOUT_DIRECTION_RTL);
```

پارامتر ورودی متد `setLayoutDirection` از نوع `View.LAYOUT_DIRECTION_RTL` است که از نام آن پیداست که `Direction` (جهت) چینش اجزای `Layout` از نوع `RTL` اعمال می‌شود.

پروژه تولبار فقط یک اکتیویتی دارد بنابراین این خط درون `MainActivity` قرار می‌گیرد.

پروژه را اجرا می‌کنم:





مانند حالتی که در مرحله قبل `android:layoutDirection="rtl"` را به `Layout` اصلی اکتیویتی اضافه کردم، با استفاده از متد `setLayoutDirection` نیز تمامی عناصر موجود در اکتیویتی به حالت راست چین تغییر وضعیت پیدا می‌کنند.

نحوهی راست چین کردن `Toolbar` در اندروید را بررسی کردیم. پیاده سازی هردو روش به سادگی امکان‌پذیر است و نیازی به اعمال تغییرات در `View` ها نیست. البته کار همیشه به همین سادگی نخواهد بود. در مواردی ممکن است نیاز به اعمال تغییرات در عناصر موجود در صفحه باشد که در وب مقالات متعددی در این خصوص وجود دارد. یا اگر تمایل دارید اپلیکیشن راست چین شما از دیوایس‌های با API پایینتر از ۱۷ نیز به خوبی پشتیبانی کند، کار مقداری پیچیده‌تر خواهد شد. به امید خدا من هم در ادامه آموزش‌ها به نحوهی راست چین کردن سایر قسمت‌های یک پروژه اندرویدی خواهم پرداخت.

**مطالعه‌ی بیشتر:**

<https://developer.android.com/reference/android/util/LayoutDirection>

با ارائه انتقادات و پیشنهادات خود، ما را در ارائه آموزش‌های بهتر یاری فرمائید.

[www.android-studio.ir](http://www.android-studio.ir)