



آموزش برنامه نویسی اندروید در محیط اندروید استودیو

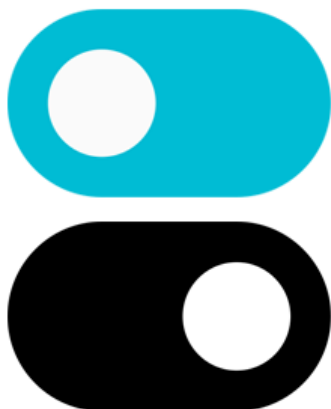
کار با دکمه Switch

مدرس : سیدمهدی مطهری

www.android-studio.ir



به نام خدا

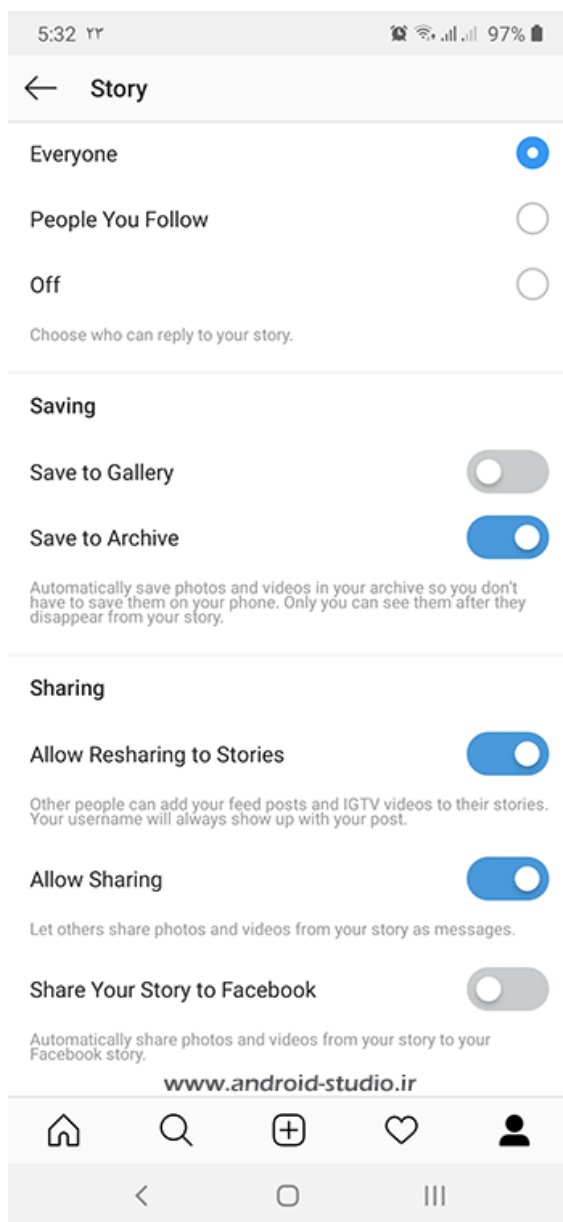


Android Switch Button
www.android-studio.ir

در این جلسه از سری مباحث **آموزش برنامه نویسی اندروید** به نحوه کار با دکمه سوئیچ یا Switch Button در اندروید می‌پردازیم. از این دکمه عموماً برای فعال/غیرفعال کردن یک قابلیت در برنامه‌ها استفاده می‌شود.

Switch Button چیست؟

اگر تابحال نام این دکمه را نشنیده‌اید، مطمئناً به مراتب و در برنامه‌های مختلفی از آن استفاده کرده‌اید. عمده کاربرد دکمه Switch در فعال (ON) و غیرفعال (OFF) کردن امکانات و قابلیت‌های موجود در اپلیکیشن‌هاست. اگر سری به بخش تنظیمات یک برنامه مانند شبکه اجتماعی اینستاگرام بزنید، تعداد زیادی Switch Button را مشاهده خواهید کرد.



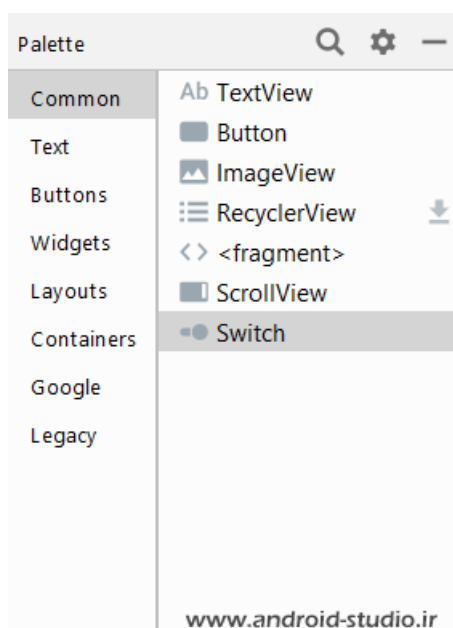
از دیگر کاربردهای این ویجت/المان می‌توان به فعال و غیر فعال کردن تم تیره (Dark theme) در برخی برنامه‌ها اشاره کرد. این ویجت در Android 4 یعنی API 14 معرفی شده بنابراین با تمامی دستگاه‌های اندرویدی موجود در بازار سازگاری کامل دارد. بیشتر از این نیاز به توضیح پیرامون کاربرد این دکمه نیست بنابراین به اصل مطلب می‌پردازم.



ساخت پروژه Switch Button در اندروید استودیو

ابتدا طبق مبحث [آموزش ساخت پروژه در اندروید استودیو](#) یک پروژه اندرویدی با نام Switch می‌سازم. اکتیویتی را از نوع Empty Activity و زبان را Java انتخاب کردم.

در قدم نخست و در layout اکتیویتی یک Switch و یک TextView اضافه می‌کنم. Switch هم مانند سایر ویجت‌ها هم به صورت دستی در حالت Text و هم از طریق پالت اندروید استودیو و به شیوه drag & drop به لایه اضافه می‌شود:



activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    tools:context=".MainActivity">

    <Switch
        android:id="@+id/switch_btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Switch" />

    <TextView
        android:id="@+id/txt_view"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```



```
        android:text="Hello World!" />
</LinearLayout>
```

در ادامه و پس از تعریف این دو ویجت در کلاس اکتیویتی، متد شنونده (Listener) برای دکمه Switch را می‌سازم:

MainActivity.java

```
package ir.android_studio.aswitch;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.CompoundButton;
import android.widget.Switch;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private Switch mSwitch;
    private TextView txtView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mSwitch = findViewById(R.id.switch_btn);
        txtView = findViewById(R.id.txt_view);

        mSwitch.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener()
        {
            @Override
            public void onCheckedChanged(CompoundButton compoundButton, boolean b) {

            }
        });
    }
}
```

برخلاف دکمه‌های معمولی که متد شنونده آن `setOnClickListener()` هست، برای Switch Button ها از `setOnCheckedChangeListener()` استفاده می‌شود. درون این متد یک متد با نام `onCheckedChanged()` توسط اندروید استودیو اضافه شد که دستورات مدنظر ما درون این متد



تعریف می‌شود. پارامتر دوم این متد از جنس boolean و با نام b نامگذاری شده که تغییر وضعیت دکمه را اعلام می‌کند. یک شرط به صورت زیر تعریف می‌کنم:

```
@Override
public void onCheckedChanged(CompoundButton compoundButton, boolean b) {

    if (b) {

        textView.setText("فعال");

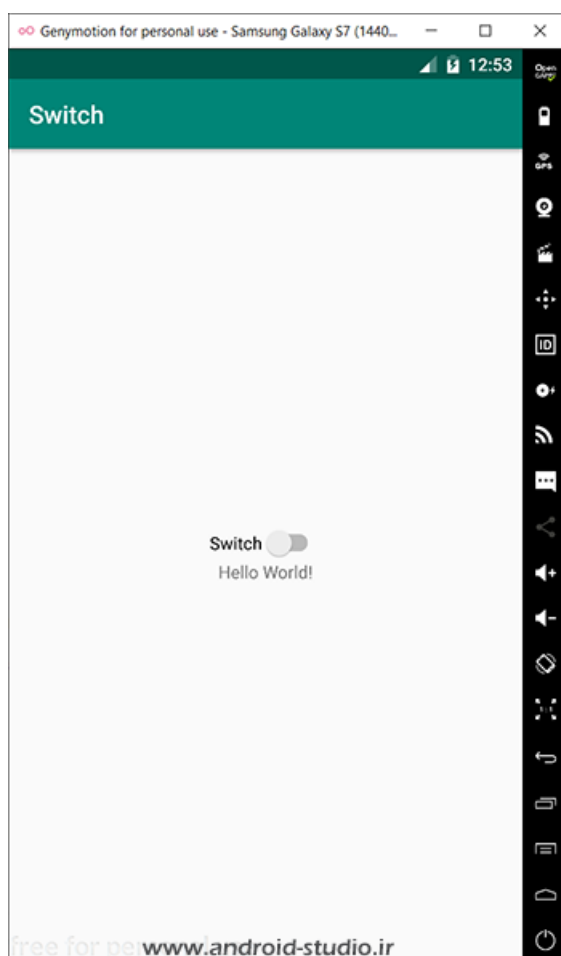
    } else {

        textView.setText("غیر فعال");

    }

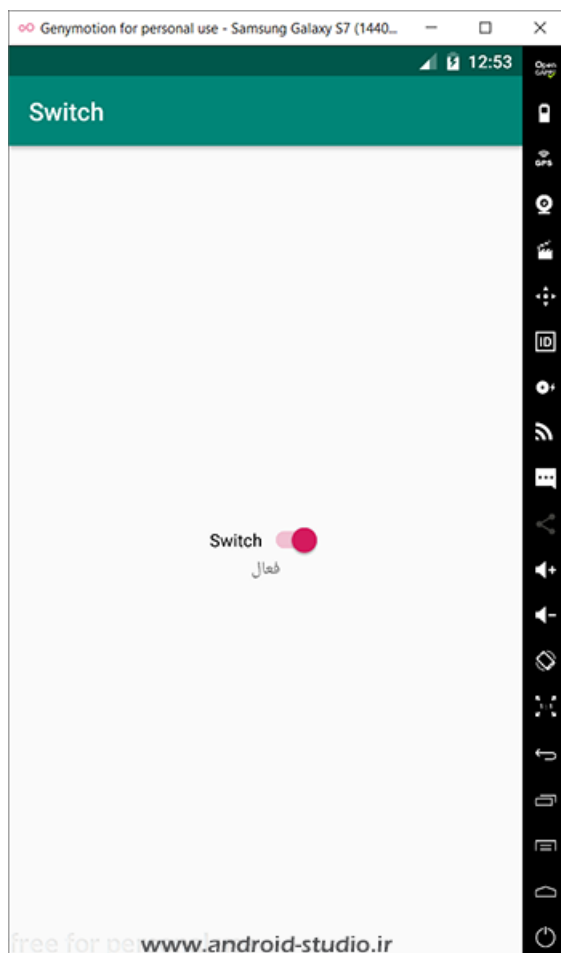
}
```

شرط فوق بررسی می‌کند چنانچه b مقدار true برگرداند یعنی دکمه در وضعیت فعال یا ON قرار گرفته باشد، قسمت اول شرط و در غیر اینصورت قسمت دوم شرط اجرا شود. پروژه را اجرا می‌کنم:

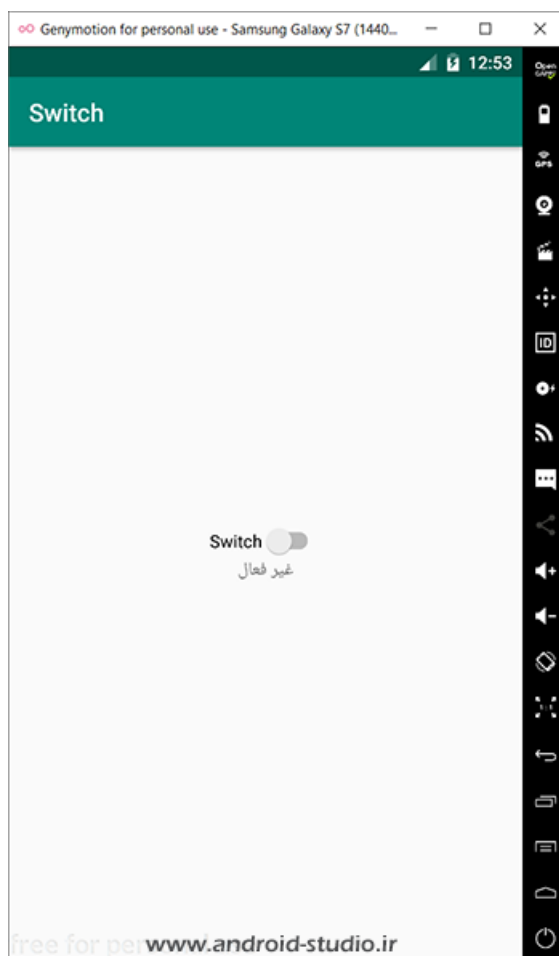




در حالت پیش فرض، دکمه در وضعیت خاموش قرار دارد. روی سوئیچ کلیک می‌کنم تا در وضعیت روشن قرار گیرد:



با قرار گرفتن وضعیت دکمه به حالت فعال یا روشن، قسمت نخست شرط اجرا و عبارت "فعال" روی TextView اعمال شد. مجدداً با تغییر وضعیت دکمه به حالت غیر فعال یا خاموش، انتظار می‌رود عبارت "غیر فعال" نمایش داده شود؛ یعنی قسمت دوم شرط اجرا شود:



در مواردی ممکن است بخواهیم خارج از متد listener دکمه سوئیچ، وضعیت خاموش یا روشن بودن آن را بررسی کنیم. در این حالت از متد `isChecked()` استفاده می‌کنیم. برای تمرین، یک Button به اکتیویتی اضافه می‌کنم:

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    tools:context=".MainActivity">

    <Switch
        android:id="@+id/switch_btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Switch"
```



```

        android:checked="true"/>

<TextView
    android:id="@+id/txt_view"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!" />

<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="وضعیت دکمه سوئیچ" />

</LinearLayout>

```

سپس در متد شنونده این دکمه یک شرط تعریف می‌کنم:

```

mButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        if (mSwitch.isChecked()) {
            Toast.makeText(MainActivity.this, "On", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(MainActivity.this, "Off", Toast.LENGTH_SHORT).show();
        }
    }
});

```

در صورتی که `mSwitch.isChecked()` مقدار `true` برگرداند (یعنی Switch Button در حالت فعال قرار گرفته) قسمت اول و در غیر اینصورت قسمت دوم شرط اجرا خواهد شد.

کد کامل اکتیویتی:

MainActivity.java

```

package ir.android_studio.aswitch;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CompoundButton;
import android.widget.Switch;
import android.widget.TextView;
import android.widget.Toast;

```



```
public class MainActivity extends AppCompatActivity {

    private Switch mSwitch;
    private TextView txtView;
    private Button mButton;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mSwitch = findViewById(R.id.switch_btn);
        txtView = findViewById(R.id.txt_view);
        mButton = findViewById(R.id.button);

        mSwitch.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener()
        {
            @Override
            public void onCheckedChanged(CompoundButton compoundButton, boolean b) {

                if (b) {

                    txtView.setText("فعال");

                } else {

                    txtView.setText("غیر فعال");

                }

            }

        });

        mButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                if (mSwitch.isChecked()) {
                    Toast.makeText(MainActivity.this, "On", Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(MainActivity.this, "Off", Toast.LENGTH_SHORT).show();
                }

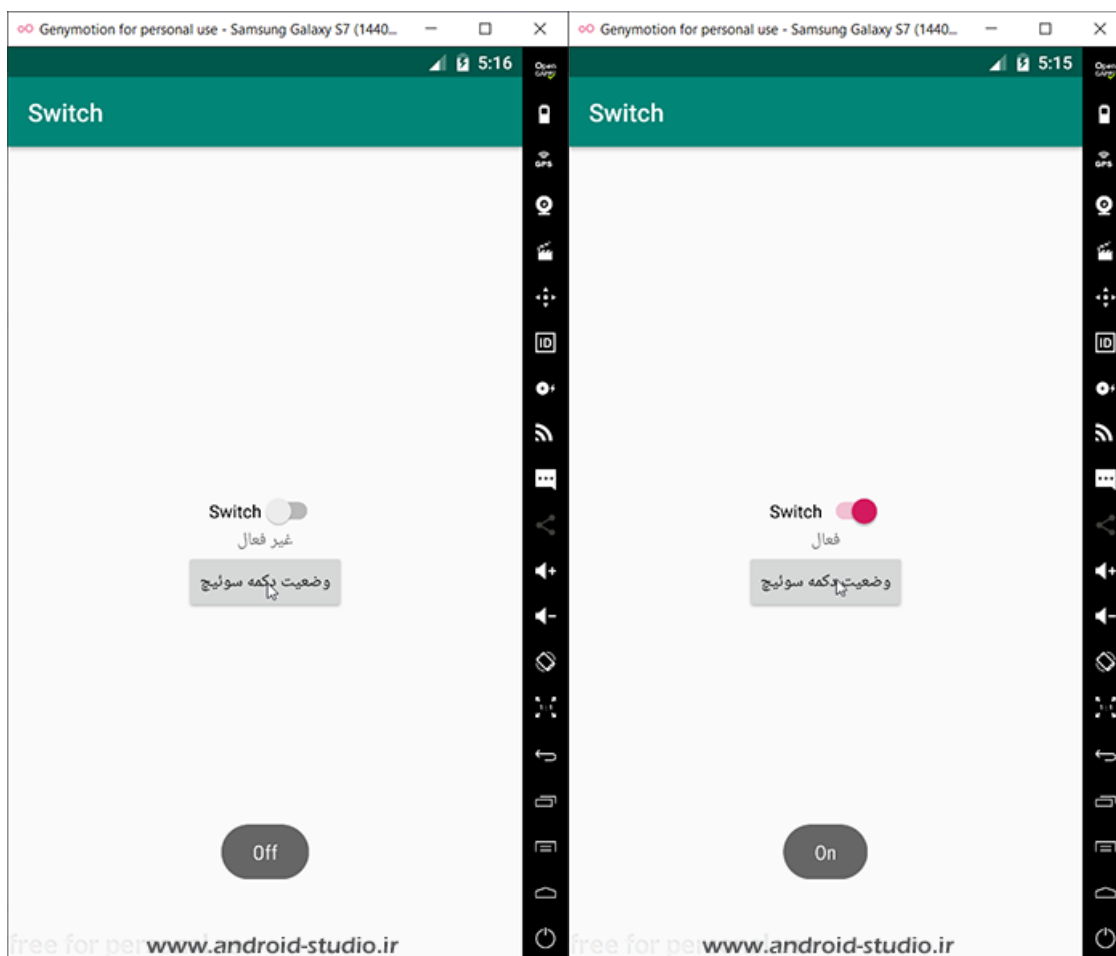
            }

        });

    }

}
```

مجدد پروژه را اجرا می‌کنم:



با استفاده از `isChecked()` هر زمان که لازم باشد می‌توانیم وضعیت دکمه سوئیچ را بررسی کنیم. برای آشنایی با نحوه استایل دهی ویجت‌ها به سبک متریال دیزاین آموزش استایل و تم متریال در اندروید را مطالعه نمایید.

موفق و پیروز باشید.

مطالعه بیشتر:

<https://developer.android.com/reference/android/widget/Switch>

توجه: سوره پروژه درون پوشه Exercises قرار دارد

با ارائه انتقادات و پیشنهادات خود، ما را در ارائه آموزش‌های بهتر یاری فرمائید.
این فایل رایگان بوده و انتشار آن (بدون دخل و تصرف) مانعی ندارد.