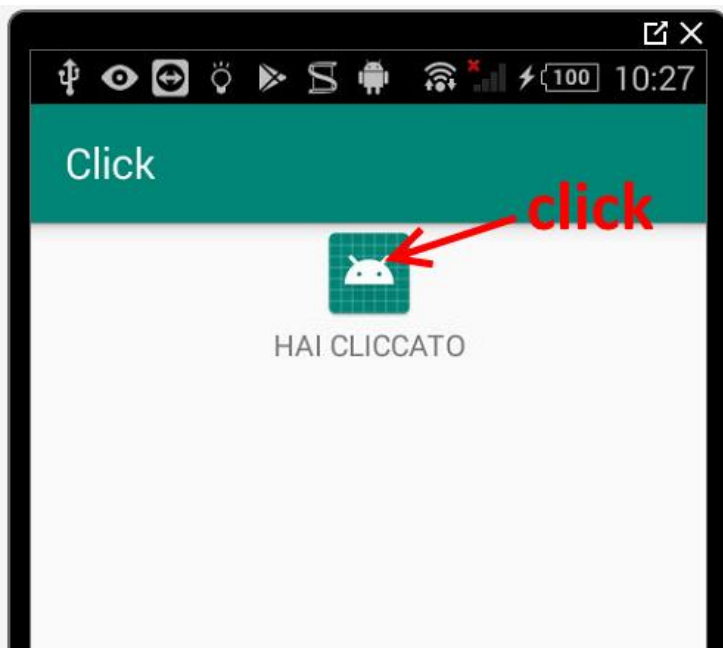


## Esempio Pilota: GestioneEventi

### activity\_main.xml - interfaccia grafica



```
ImageView imv
TextView tvMessaggio
```

### activity\_main.xml - codice XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

<TextView
    android:id="@+id/tvMessaggio"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Prova a cliccare sull'immagine"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imv" />

<ImageView
    android:id="@+id/imv"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:srcCompat="@mipmap/ic_launcher" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

### MainActivity.java - Codice JAVA

```
package itis.classe4ainf.click;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
import android.view.View;
```

```
import android.widget.ImageView;
```

```
import android.widget.TextView;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    TextView tvMessaggio;
```

```
    ImageView imv;
```

*/\* Per gestire un **Evento** (es. Click) su una **View** (es. ImageView) è necessario creare un "oggetto **LISTENER**" e **associarlo** alla View.*

*Per creare l'oggetto **LISTENER** è necessario creare una classe (es. **miaClasseListener**), che **IMPLEMENTI** una opportuna **INTERFACCIA** (es. **ImageView.OnClickListener**).*

*Una **INTERFACCIA** è un po' come una **Classe Astratta**: ha dei metodi astratti che **devono essere ridefiniti** quando la si deriva.*

*Nel nostro caso, quindi, qui **definiamo una nuova classe** che "implementa" l'interfaccia "ImageView.OnClickListener" ridefinendone il metodo "onTouch"... \*/*

```
public class miaClasseListener
    implements ImageView.OnClickListener
{
    @Override
    public void onClick(View v) {
        tvMessaggio.setText ("HAI CLICCATO");
    }
}
```

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

```
// Recuperiamo tvMessaggio (TextView)
// e imv (ImageView) dal layout ...
```

```
imv = findViewById(R.id.imv);
tvMessaggio = findViewById(R.id.tvMessaggio);
```

```
// Creiamo un oggetto (il "listener" = "ascoltatore")
// della nostra nuova classe "miaClasseListener" ...
```

```
miaClasseListener mioListener = new miaClasseListener();
```

```
// Impostiamo (o "registriamo") il nostro "listener"  
// per gestire il Click sulla nostra immagine ...
```

```
imv.setOnClickListener(mioListener);
```

*/\* NOTA BENE: in Java, questa operazione è sintetizzabile  
in un'unica istruzione così articolata ...*

```
imv.setOnClickListener  
( new ImageView.OnClickListener()  
{  
    public void onClick(View v)  
    {  
        tvMessaggio.setText("HAI CLICCATO");  
    }  
});
```

```
*/
```

```
}  
}
```