

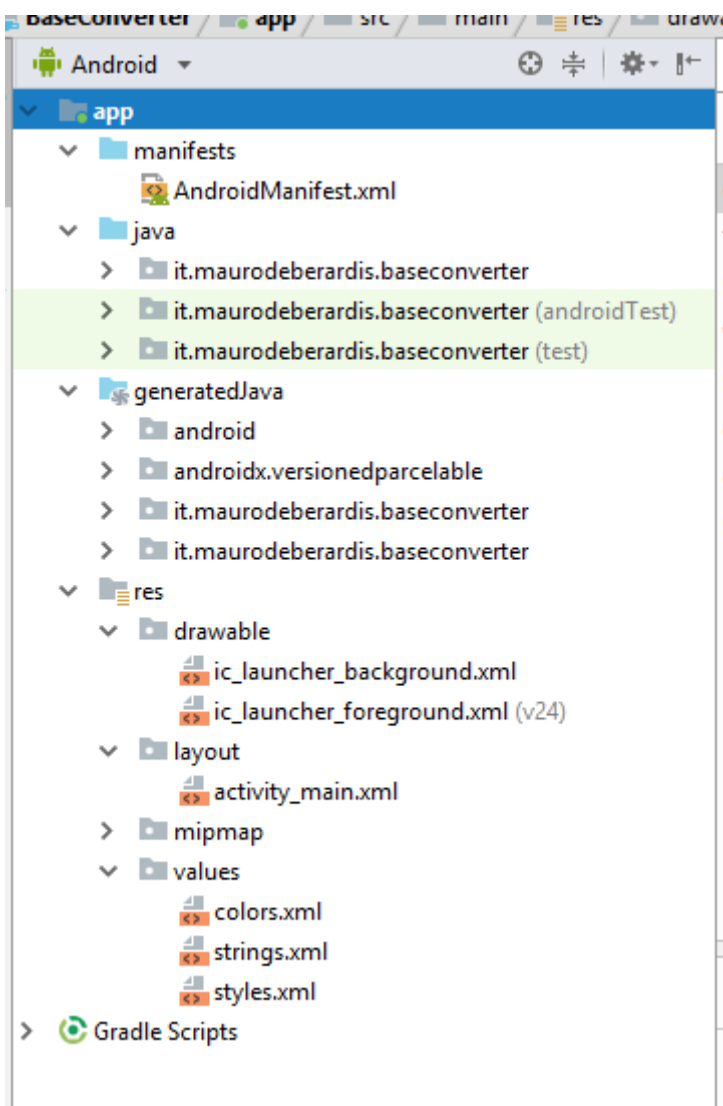
## Convertitore di basi

L'app consente di convertire numeri positivi e interi tra le basi decimale, binaria, ottale ed esadecimale. I valori accettati in input sono compresi tra 0 e  $2^{31}$  (in base dieci).

Gli studenti volenterosi e bravi possono migliorare l'app dal punto di vista estetico e, con alcune modifiche non particolarmente complesse, consentire:

1. la conversione di numeri compresi tra 0 e  $2^{63}$  (in base dieci).
2. la conversione di numeri anche non interi

### Progetto Android Studio



## 1) AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="it.maurodeberardis.baseconverter">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">

        <activity android:name=".MainActivity"
            android:configChanges="orientation"
            android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

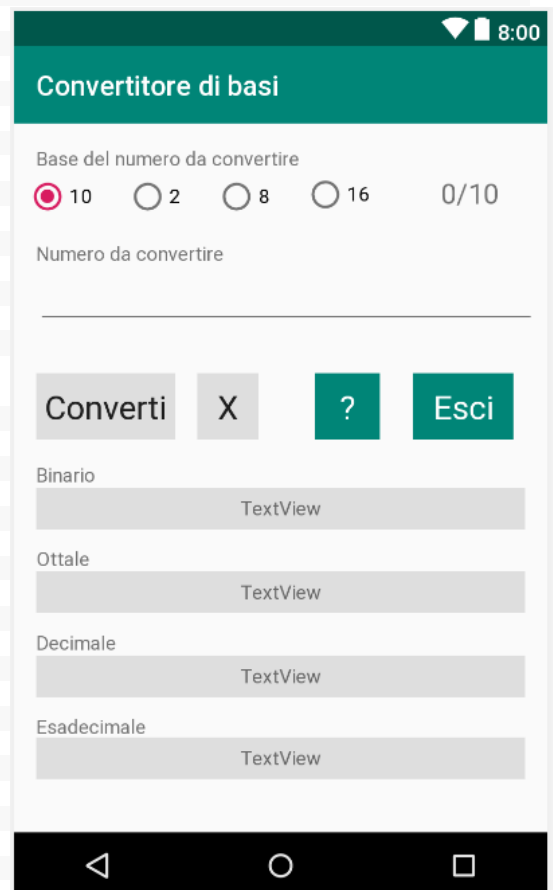
## 2) activity\_main.xml (Text e Design)

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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/textView1"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="16dp"
        android:text="Base del numero da convertire"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioButton
        android:id="@+id/radio10"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="2dp"
        android:checked="true"
        android:text="10"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView1" />

    <RadioButton
        android:id="@+id/radio2"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="24dp"
        android:layout_marginLeft="24dp"
        android:layout_marginTop="2dp"
        android:text="2"
        app:layout_constraintStart_toEndOf="@+id/radio10"
        app:layout_constraintTop_toBottomOf="@+id/textView1" />
```



```
<RadioButton
    android:id="@+id/radio8"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="24dp"
    android:layout_marginLeft="24dp"
    android:layout_marginTop="2dp"
    android:text="8"
    app:layout_constraintStart_toEndOf="@+id/radio2"
    app:layout_constraintTop_toBottomOf="@+id/textView1" />

<RadioButton
    android:id="@+id/radio16"
    android:layout_width="0dp"
    android:layout_height="29dp"
    android:layout_marginStart="24dp"
    android:layout_marginLeft="24dp"
    android:layout_marginTop="2dp"
    android:text="16"
    app:layout_constraintStart_toEndOf="@+id/radio8"
    app:layout_constraintTop_toBottomOf="@id/textView1" />

<TextView
    android:id="@+id/tConta"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginLeft="32dp"
    android:layout_marginTop="2dp"
    android:layout_marginEnd="16dp"
    android:layout_marginRight="16dp"
    android:gravity="center"
    android:text="0/10"
    android:textSize="20sp"
    android:textStyle="normal"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toEndOf="@+id/radio16"
    app:layout_constraintTop_toBottomOf="@+id/textView1" />

<TextView
    android:id="@+id/EticNumero"
    android:layout_width="0dp"
    android:layout_height="18dp"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="16dp"
    android:layout_marginEnd="16dp"
    android:layout_marginRight="16dp"
    android:text="Numero da convertire"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/radio10" />

<EditText
    android:id="@+id/numero"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:ems="10"
    android:inputType="text"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/EticNumero" />

<Button
    android:id="@+id/bConverti"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="32dp"
    android:background="#dedede"
```

```
android:text=" Converti "  
android:textAppearance="@style/TextAppearance.AppCompat.Headline"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/numero" />
```

<Button

```
android:id="@+id/bClear"  
android:layout_width="44dp"  
android:layout_height="wrap_content"  
android:layout_marginStart="16dp"  
android:layout_marginLeft="16dp"  
android:layout_marginTop="32dp"  
android:background="#dedede"  
android:text="X"  
android:textAppearance="@style/TextAppearance.AppCompat.Headline"  
app:layout_constraintStart_toEndOf="@+id/bConverti"  
app:layout_constraintTop_toBottomOf="@+id/numero" />
```

<Button

```
android:id="@+id/bInfo"  
android:layout_width="47dp"  
android:layout_height="wrap_content"  
android:layout_marginStart="32dp"  
android:layout_marginLeft="32dp"  
android:layout_marginTop="32dp"  
android:layout_marginEnd="8dp"  
android:layout_marginRight="8dp"  
android:background="#008577"  
android:text="?"  
android:textAppearance="@style/TextAppearance.AppCompat.Headline"  
android:textColor="#ffffff"  
app:layout_constraintEnd_toStartOf="@+id/bFine"  
app:layout_constraintStart_toEndOf="@+id/bClear"  
app:layout_constraintTop_toBottomOf="@+id/numero" />
```

<Button

```
android:id="@+id/bFine"  
android:layout_width="72dp"  
android:layout_height="wrap_content"  
android:layout_marginStart="8dp"  
android:layout_marginLeft="8dp"  
android:layout_marginTop="32dp"  
android:layout_marginEnd="16dp"  
android:layout_marginRight="16dp"  
android:background="#008577"  
android:text="Esci"  
android:textAppearance="@style/TextAppearance.AppCompat.Headline"  
android:textColor="#ffffff"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintStart_toEndOf="@+id/bInfo"  
app:layout_constraintTop_toBottomOf="@+id/numero" />
```

<TextView

```
android:id="@+id/tBase2"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_marginStart="16dp"  
android:layout_marginLeft="16dp"  
android:layout_marginTop="16dp"  
android:text="Binario"  
android:textSize="14sp"  
android:textStyle="normal"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/bConverti" />
```

<TextView

```
android:id="@+id/tBinario"  
android:layout_width="0dp"  
android:layout_height="30dp"  
android:layout_marginStart="16dp"  
android:layout_marginLeft="16dp"  
android:layout_marginEnd="16dp"  
android:layout_marginRight="16dp"  
android:background="#DEDEDE"  
android:gravity="center"  
android:text="TextView"
```

```
android:textSize="14sp"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.0"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/tBase2" />
```

```
<TextView  
    android:id="@+id/tBase8"  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_marginStart="16dp"  
    android:layout_marginLeft="16dp"  
    android:layout_marginTop="12dp"  
    android:text="Ottale"  
    android:textSize="14sp"  
    android:textStyle="normal"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/tBinario" />
```

```
<TextView  
    android:id="@+id/tOttale"  
    android:layout_width="0dp"  
    android:layout_height="30dp"  
    android:layout_marginStart="16dp"  
    android:layout_marginLeft="16dp"  
    android:layout_marginEnd="16dp"  
    android:layout_marginRight="16dp"  
    android:background="#DEDEDE"  
    android:gravity="center"  
    android:text="TextView"  
    android:textSize="14sp"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/tBase8" />
```

```
<TextView  
    android:id="@+id/tBase10"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_marginStart="16dp"  
    android:layout_marginLeft="16dp"  
    android:layout_marginTop="12dp"  
    android:text="Decimale"  
    android:textSize="14sp"  
    android:textStyle="normal"  
    app:layout_constraintBottom_toTopOf="@+id/tDecimale"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/tOttale" />
```

```
<TextView  
    android:id="@+id/tDecimale"  
    android:layout_width="0dp"  
    android:layout_height="30dp"  
    android:layout_marginStart="16dp"  
    android:layout_marginLeft="16dp"  
    android:layout_marginEnd="16dp"  
    android:layout_marginRight="16dp"  
    android:background="#DEDEDE"  
    android:gravity="center"  
    android:text="TextView"  
    android:textSize="14sp"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/tBase10" />
```

```
<TextView  
    android:id="@+id/tBase16"  
    android:layout_width="wrap_content"  
    android:layout_height="16dp"  
    android:layout_marginStart="16dp"  
    android:layout_marginLeft="16dp"  
    android:layout_marginTop="12dp"  
    android:text="Esadecimale"  
    android:textSize="14sp"
```

App realizzata in Java, ambiente Android Studio, per convertire i numeri interi tra le basi 2, 8, 10, 16 .  
Prof. Mauro De Berardis Teramo

6

```
android:textStyle="normal"  
app:layout_constraintBottom_toTopOf="@+id/tEsadecimale"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/tDecimale" />
```

```
<TextView  
    android:id="@+id/tEsadecimale"  
    android:layout_width="0dp"  
    android:layout_height="30dp"  
    android:layout_marginStart="16dp"  
    android:layout_marginLeft="16dp"  
    android:layout_marginTop="1dp"  
    android:layout_marginEnd="16dp"  
    android:layout_marginRight="16dp"  
    android:background="#DEDEDE"  
    android:gravity="center"  
    android:text="TextView"  
    android:textSize="14sp"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/tBase16" />
```

```
</android.support.constraint.ConstraintLayout>
```

### 3) Modifica di **strings.xml**

```
<resources>  
    <string name="app_name">Convertitore di basi</string>  
</resources>
```

#### 4) MainActivity.java

```
package it.maurodeberardis.baseconverter;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.graphics.Color;
import android.support.v4.app.NotificationCompatSideChannelService;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.Spanned;
import android.view.View;
import android.text.Editable;
import android.text.InputFilter;
import android.text.TextWatcher;
import android.view.inputmethod.InputMethodManager;
import android.widget.FrameLayout;
import android.widget.RadioButton;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import static android.text.InputType.TYPE_CLASS_NUMBER;
import static android.text.InputType.TYPE_TEXT_FLAG_CAP_CHARACTERS;

public class MainActivity extends AppCompatActivity {
    private EditText numero ;
    private Button converti, clear, info,fine;
    private TextView bin, dec, oct, esa,conta;
    private TextView etic2,etic8,etic10,etic16,eticnumero;
    private RadioButton radio2,radio8,radio10,radio16;
    private double BaseFrom;
    String vp;
    private int maxcifre,MAXNUMERO;
    final Context context = this;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        BaseFrom=10.0;
        vp="";
        MAXNUMERO=2147483647;
        /*
        MAXNUMERO è il massimo numero intero in base 10 che può essere convertito dall'app.
        Java utilizza per gli interi 32 bit, il primo é per il segno, gli altri 31 per il numero.
        2^31=2147483647 (2Giga). Di conseguenza in base 2 il massimo numero convertibile è di 31 bit a 1,
        in base 8 è 17777777777, in base 16 7FFFFFFF.
        La limitazione è dovuta al fatto che per la conversione da decimale a base 2, o 8 o 16, si utilizza
        l'operatore % (resto della divisione) che funziona con numeri interi.
        Ovviamente non sarebbe difficile, rinunciando all'utilizzo dell'operatore %, utilizzare
        numeri double a 64 bit al posto dei numeri int e avere la possibilità di convertire
        numeri, delle 4 basi, molto più grandi (da 0 a 263).
        */
        numero = (EditText) findViewById(R.id.numero);
        numero.addTextChangedListener(new MyTextWatcher(numero));
        conta=(TextView)findViewById(R.id.tConta);
        numero.setInputType(TYPE_CLASS_NUMBER);//setta la tastiera numerica
        maxcifre=10;
        eticnumero=(TextView) findViewById(R.id.EticNumero);
        eticnumero.setText("Numero in base 10 da convertire");
        converti = (Button) findViewById(R.id.bConverti);
        clear=(Button) findViewById(R.id.bClear);
        info=(Button)findViewById(R.id.bInfo);
        fine=(Button)findViewById(R.id.bFine);
        // etichette per il risultato delle conversioni
        etic2=(TextView) findViewById(R.id.tBase2);
        etic8=(TextView) findViewById(R.id.tBase8);
        etic10=(TextView) findViewById(R.id.tBase10);
        etic16=(TextView) findViewById(R.id.tBase16);
        //TextView per i risultati della conversione
        bin=(TextView) findViewById(R.id.tBinario);
        dec=(TextView) findViewById(R.id.tDecimale);
        oct=(TextView) findViewById(R.id.tOttale);
        esa=(TextView) findViewById(R.id.tEsadecimale);
        displaytext(0);
    }
}
```

```
//RadioButton che consentono di scegliere la base da cui convertire
radio2=(RadioButton) findViewById(R.id.radio2);
radio8=(RadioButton) findViewById(R.id.radio8);
radio10=(RadioButton) findViewById(R.id.radio10);
radio16=(RadioButton) findViewById(R.id.radio16);
conta.setText("0/10");

/* se si clicca su radio2 vengono "unchecked" tutti i RadioButton tramite la funzione
   resetradiobutton() e viene settato radio2. L'operazione è inutile se si usa un controllo
   RadioGroup contenente i 4 RadioButton. Stessa considerazione per radio8,radio10,radio16 */
radio2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        resetradiobutton();
        radio2.setChecked(true);
        BaseFrom=2.0;maxcifre=31;
        String msg="Numero in base 2 da convertire";
        numero.setInputType(TYPE_CLASS_NUMBER);//setta la tastiera numerica
        eticnumero.setText(msg);
        String contatore="0/"+String.valueOf(maxcifre);
        conta.setText(contatore);
        displaytext(0);
    }
});

radio8.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        resetradiobutton();
        radio8.setChecked(true);
        BaseFrom=8.0;maxcifre=11;
        String msg="Numero in base 8 da convertire";
        numero.setInputType(TYPE_CLASS_NUMBER);//setta la tastiera numerica
        eticnumero.setText(msg);
        String contatore="0/"+String.valueOf(maxcifre);
        conta.setText(contatore);
        displaytext(0);
    }
});

radio10.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        resetradiobutton();
        radio10.setChecked(true);
        BaseFrom=10.0;maxcifre=10;
        numero.setInputType(TYPE_CLASS_NUMBER);//setta la tastiera numerica
        String msg="Numero in base 10 da convertire";
        numero.setInputType(TYPE_CLASS_NUMBER);//setta la tastiera numerica
        eticnumero.setText(msg);
        String contatore="0/"+String.valueOf(maxcifre);
        conta.setText(contatore);
        displaytext(0);
    }
});

radio16.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        resetradiobutton();
        radio16.setChecked(true);
        BaseFrom=16.0;maxcifre=8;
        numero.setInputType(TYPE_TEXT_FLAG_CAP_CHARACTERS);//setta la tastiera alfanumerica
        String msg="Numero in base 16 da convertire";
        eticnumero.setText(msg);
        String contatore="0/"+String.valueOf(maxcifre);
        conta.setText(contatore);
        displaytext(0);
    }
});
});
```



```
/* Quando si clicca sul bottone "X", viene richiamata la funzione clearall() che
pulisce il numero fornito in input e i risultati delle conversioni di base */
clear.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        clearall();
    }
});

/* Quando si clicca sul bottone "Converti" viene richiamata la funzione convertfrombase() che
converte il numero fornito in input, dalla base selezionata con RadioButton alle altre basi */
converti.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        convertfrombase();
    }
});

/* Quando si clicca sul bottone "?" viene aperta una finestra di dialogo di informazioni */
info.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View arg0) {
        String info="by Mauro De Berardis @2019\n";
        info+="\nConverte numeri interi positivi\n";
        info+="\nMax numero bin: 31bit 1";
        info+="\nMax numero oct: 1777777777";
        info+="\nMax numero dec: 2147483647";
        info+="\nMax numero esa: 7FFFFFFF";
        AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(context);
        alertDialogBuilder.setTitle("Convertitore di basi");
        alertDialogBuilder
            .setMessage(info)
            .setCancelable(false)
            .setNegativeButton("Ok",new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int id) {
                    dialog.cancel();
                }
            });
        AlertDialog alertDialog = alertDialogBuilder.create();
        alertDialog.show();
    }
});

/* Quando si clicca sul bottone "Esci" l'app viene chiusa */
fine.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        finish();
        System.exit(0);
    }
});

} //chiude onCreate()-----

private void resetradiobutton();//uncheck di tutti i RadioButton
{
    radio2.setChecked(false);
    radio8.setChecked(false);
    radio10.setChecked(false);
    radio16.setChecked(false);
    numero.setText(""); // quando si cambia la base il numero in input viene resettato
}

private void clearall();//pulisce i campi Text e nasconde i campi dei risultati
{
    numero.setText("");
    bin.setText("");
    oct.setText("");
    dec.setText("");
    esa.setText("");
    displaytext(0);
    numero.setError("");
    numero.setError(null);
}
}
```

```
private void convertfrombase()
{
    if (numero.getText().toString().trim().isEmpty()) {
        numero.setError("nessun numero inserito");
        numero.requestFocus();
        return;
    }

    String v = numero.getText().toString();

    if (BaseFrom==10.0)
    {
        if(Double.parseDouble(v)>2147483647) // se il valore decimale è superiore a 2^31
        {
            numero.setError("Max numero decimale: "+String.valueOf(MAXNUMERO));
            numero.requestFocus();
            return;
        }
        int decimale= Integer.parseInt(v);
        dec.setText(String.valueOf(decimale));
        bin.setText(convertfrombase10(decimale,2));
        oct.setText(convertfrombase10(decimale,8));
        esa.setText(convertfrombase10(decimale,16));
    }

    if (BaseFrom==2.0){
        //il controllo sul valore del numero viene fatto controllando la lunghezza della stringa,
        // che viene fatta in TextWatcher, che non può essere superiore a 31
        int decimale=(int)converttobase10();
        dec.setText(String.valueOf(decimale));
        bin.setText(convertfrombase10(decimale,2));
        oct.setText(convertfrombase10(decimale,8));
        esa.setText(convertfrombase10(decimale,16));
    }

    if (BaseFrom==8.0){

        if(converttobase10(>MAXNUMERO)// se il valore decimale è superiore a 2^31
        {
            numero.setError("Max numero ottale: 1777777777");
            numero.requestFocus();
            return;
        }
        int decimale=(int) converttobase10();
        dec.setText(String.valueOf(decimale));
        bin.setText(convertfrombase10(decimale,2));
        oct.setText(convertfrombase10(decimale,8));
        esa.setText(convertfrombase10(decimale,16));
    }
    /* Nel caso della base 16 il controllo dei simboli viene fatto sull'evento onClick()
    del button converti. Non ho utilizzato la classe inputFilter che all'interno di textWatcher
    funziona perfettamente nel filtraggio di valori numerici (e quindi per le basi 2, 8 e 10) e
    del limite massimo del numero fornito in input, ma non ha funzionato per quel che riguarda
    il filtraggio di valori alfanumeri. Non ho insistito nel risolvere il problema riscontrato,
    qualche studente volenteroso può provare a farlo ☺
    */
    if (BaseFrom==16.0){
        if(!v.matches("[a-fA-F0-9]*$")){
            numero.setError("Simboli ammessi da 0 a 9 e da A a F");
            numero.requestFocus();
            return;
        }
    }

    if(converttobase10(>MAXNUMERO)// se il valore decimale è superiore a 2^31
    {
        numero.setError("Max numero esadecimale: 7FFFFFFF");
        numero.requestFocus();
        return;
    }
    int decimale=(int) converttobase10();
    dec.setText(String.valueOf(decimale));
    bin.setText(convertfrombase10(decimale,2));
    oct.setText(convertfrombase10(decimale,8));
    esa.setText(convertfrombase10(decimale,16));
}
```

```
closekb(); //chiudo la tastiera
settaicolori();
displaytext(1);
} //chiude convertfrombase

private void displaytext(int q)
{
    //q=0 nasconde q=1 mostra
    String colore="#000000";
    if(q==0) colore="#FFFFFF";
    etic2.setTextColor(Color.parseColor(colore));
    etic8.setTextColor(Color.parseColor(colore));
    etic10.setTextColor(Color.parseColor(colore));
    etic16.setTextColor(Color.parseColor(colore));
    bin.setTextColor(Color.parseColor(colore));
    oct.setTextColor(Color.parseColor(colore));
    dec.setTextColor(Color.parseColor(colore));
    esa.setTextColor(Color.parseColor(colore));
    if(q==0) {
        colore="#FFFFFF";
        bin.setBackgroundColor(Color.parseColor(colore));
        oct.setBackgroundColor(Color.parseColor(colore));
        dec.setBackgroundColor(Color.parseColor(colore));
        esa.setBackgroundColor(Color.parseColor(colore));
    }
}

private void closekb()
{
    View view=this.getCurrentFocus();
    InputMethodManager mm=(InputMethodManager) getSystemService(Context.INPUT_METHOD_SERVICE);
    mm.hideSoftInputFromWindow(view.getWindowToken(),0);
}

private double converttobase10() {
    String v = numero.getText().toString();
    double b=BaseFrom;
    int l=v.length();
    double c=0, somma=0;

    if (b==2.0 || b==8.0)
    {
        for(int i=0;i<l;i++)
        {
            String ch=v.substring(i,i+1);
            c=Double.parseDouble(ch);
            somma+=c*Math.pow(b,l-i-1);
        }
    }
    if(b==16.0)
    {
        for(int i=0;i<l;i++)
        {
            String ch=v.substring(i,i+1);

            if(ch.equals("A") || ch.equals("a"))
                c=10.0;
            else if(ch.equals("B") || ch.equals("b"))
                c=11.0;
            else if(ch.equals("C") || ch.equals("c"))
                c=12.0;
            else if(ch.equals("D") || ch.equals("d"))
                c=13.0;
            else if(ch.equals("E") || ch.equals("e"))
                c=14.0;
            else if(ch.equals("F") || ch.equals("f"))
                c=15.0;
            else
                c=Double.parseDouble(ch);

            somma+=c*Math.pow(b,l-i-1);
        }
    }
    return somma;
}
```

```
private String convertfrombase10( int num, int b)
{
    String risultato = "";
    int resto;
    while (num > 0) {
        resto = num % b;
        if (b == 16) {
            if (resto == 10)
                risultato = "A"+risultato;
            else if (resto == 11)
                risultato = "B"+risultato;
            else if (resto == 12)
                risultato = "C"+risultato;
            else if (resto == 13)
                risultato = "D"+risultato;
            else if (resto == 14)
                risultato = "E"+risultato;
            else if (resto == 15)
                risultato = "F"+risultato;
            else
                risultato = resto+risultato;
        } else
            risultato= resto+risultato;
        num /= b;
    }
    return risultato;
}

private void settaicolori()
{
    bin.setBackgroundColor(Color.parseColor("#DEDEDE"));
    oct.setBackgroundColor(Color.parseColor("#DEDEDE"));
    dec.setBackgroundColor(Color.parseColor("#DEDEDE"));
    esa.setBackgroundColor(Color.parseColor("#DEDEDE"));
    if(BaseFrom==2) bin.setBackgroundColor(Color.parseColor("#00FF00"));
    if(BaseFrom==8) oct.setBackgroundColor(Color.parseColor("#00FF00"));
    if(BaseFrom==10) dec.setBackgroundColor(Color.parseColor("#00FF00"));
    if(BaseFrom==16) esa.setBackgroundColor(Color.parseColor("#00FF00"));
}

/* EditText utilizza l'interfaccia TextWatcher per gestire i cambiamenti.
Per fare questo EditText chiama il metodo addTextChangedListener().
[vedasi sopra la riga: numero.addTextChangedListener(new MyTextWatcher(numero));]
*/
private class MyTextWatcher implements TextWatcher {
    private View view;
    private MyTextWatcher(View view) {
        this.view = view;
    }
    public void beforeTextChanged(CharSequence charSequence, int i, int i1, int i2) {
    }
    public void onTextChanged(CharSequence charSequence, int i, int i1, int i2) {
    }
    @Override
    public void afterTextChanged(Editable editable) {
        // Se la prima cifra del numero da convertire vale 0, essendo non significativa,non viene accettata
        if(numero.getText().toString().equals("0")){
            numero.setText("");
            numero.requestFocus();
            return;
        }
        controllanumero();
    }
}

private void controllanumero() {
    String s = numero.getText().toString();
    // viene aggiornato il contatore di cifre inserite
    String contatore=String.valueOf(s.length())+"/"+String.valueOf(maxcifre);
    conta.setText(contatore);
    /* Il controllo della massima lunghezza del numero in input viene fatto da
    InputFilter.LengthFilter(maxcifre)
    Il controllo dei simboli inseriti nel caso delle basi 2 e 8 viene fatto da
    InputFilter.filterasimboli, nel caso della base 10 viene fatto dalla keyboard software
    TYPE_CLASS_NUMBER. La cifra digitata se non fa parte della base scelta viene man mano rifiutata.
    */
}
```

In base 16 non ho utilizzato la classe `inputFilter` `filtrasimboli`, come per le basi 2 e 8, in quanto si sono verificati problemi di malfunzionamento nella gestione della tastiera alfanumerica software `TYPE_TEXT_FLAG_CAP_CHARACTERS` (problemi forse dovuti ad un bug del software android o che, più probabilmente ☹, devono essere maggiormente approfonditi...)  
In base 16, pertanto, il controllo dei simboli non viene fatto all'atto della digitazione, ma sull'evento `onClick()` del button `converti`: l'app analizza il numero fornito in `input` e se ci sono simboli che non appartengono alla base 16, segnala l'errore e blocca la conversione.

```
*/
InputFilter filtrasimboli= new InputFilter()
{
    @Override
    public CharSequence filter(CharSequence source, int start, int end, Spanned dest, int dstart, int dend)
    {
        boolean ok;

        for (int i = start; i < end; i++)
        {
            char c = source.charAt(i);
            String ch=String.valueOf(c);
            ok=true;
            if(BaseFrom==2.0)
            {
                if (c!='0' && c!='1')ok=false;
            }
            if(BaseFrom==8.0)
            {
                if(!ch.matches(".*[0-7]"))ok=false;
            }
            if(!ok) return "";
        }
        return null;
    }
};
numero.setFilters(new InputFilter[] { filtrasimboli, new InputFilter.LengthFilter(maxcifre)});
numero.requestFocus();
}

} //chiude mainActivity-----
```

## Test dell'applicazione

L'app è stata installata su uno smartphone con schermo da 5.1, SO Android 8.0.0

