

# How to Create EditText with Cross Icon in Android

To make EditText with clear icon to delete what is written into it, we can use following codes:

**act\_layout\_search.xml**

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="fill_parent"
4      android:layout_height="wrap_content"
5      android:layout_margin="5dp"
6      android:orientation="vertical">
7      <LinearLayout android:layout_width="match_parent" android:layout_height="wrap_content">
8          <EditText
9              android:id="@+id/txt_search"
10             android:layout_width="0dp"
11             android:layout_height="40dp"
12             android:layout_weight=".85"
13
14             android:drawableRight="@drawable/ic_action_clear_edit_text"
15             android:hint="@string/search_word" >
16      </EditText>
17  </LinearLayout>
18 </RelativeLayout>
```

**MainActivity Class**

```
public class MainActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.act_layout_search.xml);
        EditText sSearchText = ((EditText) actionView.findViewById(R.id.txt_search));
        sSearchText.setOnTouchListener(new View.OnTouchListener() {
            @Override
            public boolean onTouch(View v, MotionEvent event) {
                final int DRAWABLE_LEFT = 0;
                final int DRAWABLE_TOP = 1;
                final int DRAWABLE_RIGHT = 2;
                final int DRAWABLE_BOTTOM = 3;

                if (event.getAction() == MotionEvent.ACTION_UP) {
                    Drawable
drawableClearEditText=sSearchText.getCompoundDrawables()[DRAWABLE_RIGHT];
                    if(drawableClearEditText!=null)
                        if (event.getRawX() >= (sSearchText.getRight() -
drawableClearEditText.getBounds().width())) {
                            sSearchText.setText(null);
                            return true;
                        }
                }
                return false;
            }
        });
        sSearchText.setCompoundDrawablesWithIntrinsicBounds(0, 0, 0, 0);

        sSearchText.setHint("Search in " + getIntent().getStringExtra("description") + "...");
        sSearchText.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(CharSequence charSequence, int i, int i2, int i3) {
```

```

    }

    @Override
    public void onTextChanged(CharSequence charSequence, int i, int i2, int i3) {
        if(charSequence.length()!=0){
            sSearchText.setCompoundDrawablesWithIntrinsicBounds(0, 0,
R.drawable.ic_action_clear_edit_text, 0);
        }else{
            sSearchText.setCompoundDrawablesWithIntrinsicBounds(0, 0, 0, 0);
        }

        adapter.getFilter().filter(charSequence);
    }

    @Override
    public void afterTextChanged(Editable editable) {
    }
    });
}
}
}

```