

1 UI-Elemente

1.1 TextView (Label)

```
<TextView  
    android:id="@+id/textview1"          ID  
    android:layout_width="match_parent"  match_parent oder wrap_content  
    android:layout_height="wrap_content"  
    android:text="Eingabe"  
    android:text="@string/button_finish"  aus Resource  
    android:ellipsize="marquee"  
>
```

fontFamily

typeface normal, monospace

textSize 14sp

lineSpaceExtra none, 8sp

textColor

textAlignment viewStart, textStart, center, textEnd, viewEnd

Padding

Margin

background

foreground

lines

minlines

maxlines

1.2 EditText

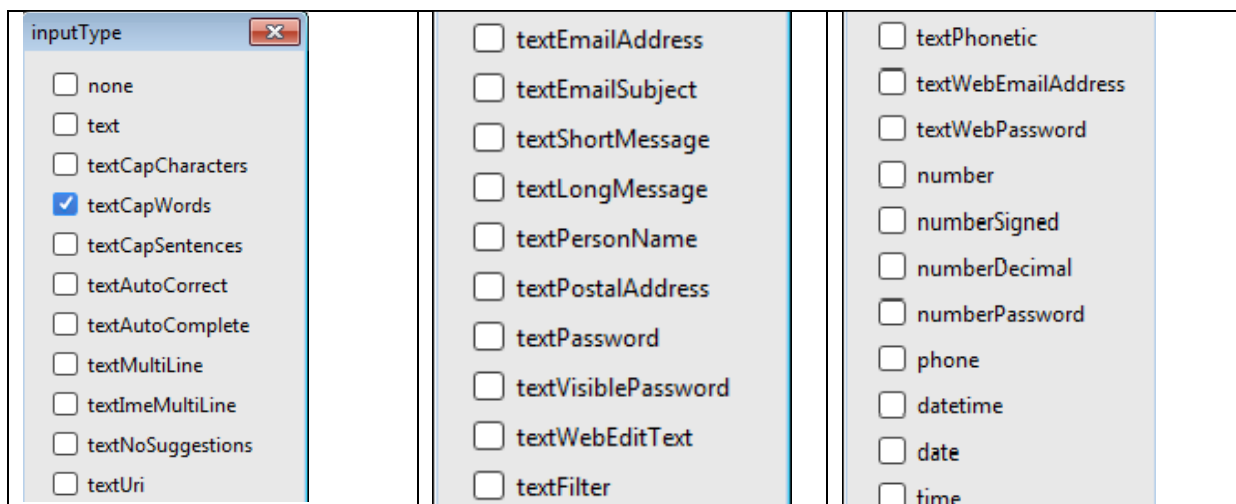
<EditText

```

    android:id="@+id/edittext1"           ID
    android:layout_width="match_parent"   match_parent oder wrap_content
    android:layout_height="wrap_content"
    android:lines="1"
    android:inputType="textCapWords"
    android:text="abc" />

```

inputType:



hint selectAllOnFocus fontFamily typeface normal, monospace textSize 14sp lineHeightExtra none, 8sp textColor textAlignment viewStart, textStart, center, textEnd, viewEnd Padding Margin	background foreground singleLine lines minlines maxlines numeric <ul style="list-style-type: none"> • integer • signed • decimal visibility
---	--

```
final EditText edittext1 = (EditText) findViewById(R.id.edittext1);
```

1.3 Button

```
<Button
```

```
    android:id="@+id/button1"
```

```
    android:text="Ok"
```

```
    android:layout_width="wrap_content"           match_parent oder wrap_content
```

```
    android:layout_height="wrap_content"
```

```
/>
```

style

- Widget.Button
- Widget.Button.Inset
- Widget.Button.Small
- Widget.Button.Toggle
- Widget.Button.Transparent
- Widget.DeviceDefault.Button
- Widget.DeviceDefault.Button.Borderless

background		lineSpaceExtra	none, 8sp
foreground		elevation	8dp
text		visibility	
fontFamily		singleLine	
typeface	normal, monospace	lines	
textSize	14sp	minlines	
textColor		maxlines	
textAlignment		Padding	
viewStart, textStart, center, textEnd, viewEnd		Margin	

```
final Button bn1 = (Button) findViewById(R.id.button1);

bn1.setOnClickListener(new View.OnClickListener() {

    @Override

    public void onClick(View v) {

        editText1.setText( "abc" );

    }

});
```

1.4 CheckBox

```
<CheckBox  
  
    android:id="@+id/chk1"  
  
    android:layout_width="match_parent"  
  
    android:layout_height="wrap_content"  
  
    android:text="Ich bin eine CheckBox"  
  
>
```

background		visibility
foreground		checked
text		clickable
fontFamily		
typeface	normal, monospace	
textSize	14sp	onClick
lineSpaceExtra	none, 8sp	
textColor		
textAligment		
	viewStart, textStart, center, textEnd, viewEnd	
Padding		
Margin		

```
final CheckBox chk1 = (CheckBox)findViewById(R.id.chk1);
```

1.5 RadioButton

```
<RadioButton
```

```
    android:id="@+id/rbAdd"
```

```
    android:layout_width="match_parent"                match_parent oder wrap_content
```

```
    android:layout_height="wrap_content"
```

```
    android:text="Add"
```

```
/>
```

background foreground text fontFamily typeface normal, monospace textSize 14sp lineSpaceExtra none, 8sp textColor textAligment viewStart, textStart, center, textEnd, viewEnd Padding Margin	visibility checked clickable onClick
---	---

```

<RadioGroup
    android:id="@+id/rbgroup"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">

    <RadioButton
        android:id="@+id/rbAdd"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Add"
        android:checked="true" />
    <RadioButton
        android:id="@+id/rbSub"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Sub"
    />
</RadioGroup>

```

```

final RadioGroup rbgroup = (RadioGroup)findViewById(R.id.rbgroup);
bn1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        switch (rbgroup.getCheckedRadioButtonId()) {
            case R.id.rbAdd:
                editText1.setText("add is selected");
                break;
            case R.id.rbSub:
                editText1.setText("sub is selected");
                break;
            default:
                editText1.setText("weder add noch sub is selected");
                break;
        }
    }
});

```

1.6 Switch

```
<Switch  
    android:id="@+id/switch1"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
/>
```

background		visibility
foreground		checked
text		clickable
fontFamily		Padding
typeface	normal, monospace	Margin
textSize	14sp	onClick
lineSpaceExtra	none, 8sp	
textColor		
textAligment		
	viewStart, textStart, center, textEnd, viewEnd	

text wird links vom Switch geschrieben !

```
final switch1 = (Switch)findViewById(R.id.switch1);
```

1.7 Spinner

```
<Spinner  
    android:id="@+id/spinner1"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:prompt="hier ist ein Spinner"  
>
```

spinnermode:

- none
- dialog
- dropdown

background		visibility
foreground		clickable
fontFamily		Padding
typeface	normal, monospace	Margin
textColor		onClick
textAligment	viewStart, textStart, center, textEnd, viewEnd	

text wird links vom Switch geschrieben !

```
final switch1 = (Switch)findViewById(R.id.switch1);
```


Java:

```
// Spinner
import java.util.ArrayList;
import java.util.List;
import android.widget.AdapterView;

spinner1 = (Spinner) findViewById(R.id.spinner1);

List<String> list = new ArrayList<String>();

    list.add("Android");

    list.add("IOS");

    list.add("Swing");

    list.add("WPF");

    list.add("Python TKinter");

ArrayAdapter<String> dataAdapter =

        new ArrayAdapter<String> (this, android.R.layout.simple_spinner_item,list);

dataAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

spinner1.setAdapter(dataAdapter);
```

Abfrage:

```
spinner1.getSelectedItem()
```

2 LinearLayout

2.1 vertikal

```
<?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:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    >

    <TextView
        android:id="@+id/textview1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Eingabe"
        android:textAlignment="viewEnd" />

    <Button
        android:id="@+id/button1"
        android:text="Aktion"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:elevation="0dp" />

</LinearLayout >
```

2.2 horizontal

```
<LinearLayout
  android:layout_width="match_parent"
  android:layout_height="wrap_content">
  android:orientation="horizontal">
  <TextView
    android:id="@+id/textview2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Switch für das Ende der Vorlesung"
    android:textAlignment="viewEnd" />
  <Switch
    android:id="@+id/switch1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="ssssssss" />

</LinearLayout>
```

3 toast

```
import android.widget.Toast;
```

```
Toast.makeText(MainActivity.this,  
    "Dies ist eine Meldung : " +  
    "\n"  
    "Zweite Zeile" ,  
    Toast.LENGTH_LONG).show(); // LENGTH_SHORT
```