



Date, Time, Tab

🕒 Created	@September 10, 2023 2:52 PM
🏷 Tags	

- Date & Time

```
<TextView
    android:id="@+id/lblDateAndTime"
    android:layout_width="fill_parent"
    android:layout_height="47px"
    android:background="#ff000099"
    android:textStyle="bold"
>
</TextView>
<Button
    android:id="@+id/btnDate"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Set the Date"
>
</Button>
<Button
    android:id="@+id/btnTime"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Set the Time"
>
</Button>
```

```
DateFormat fmtDateAndTime = DateFormat.getDateTimeInstance();
TextView lblDateAndTime;
Calendar myCalendar = Calendar.getInstance();

DatePickerDialog.OnDateSetListener d = new DatePickerDialog.OnDateSetListener()
{
    public void onDateSet(DatePicker view,
        int year, int monthOfYear, int dayOfMonth) {
        myCalendar.set(Calendar.YEAR, year);
        myCalendar.set(Calendar.MONTH, monthOfYear);
        myCalendar.set(Calendar.DAY_OF_MONTH, dayOfMonth);
        updateLabel();
    }
};

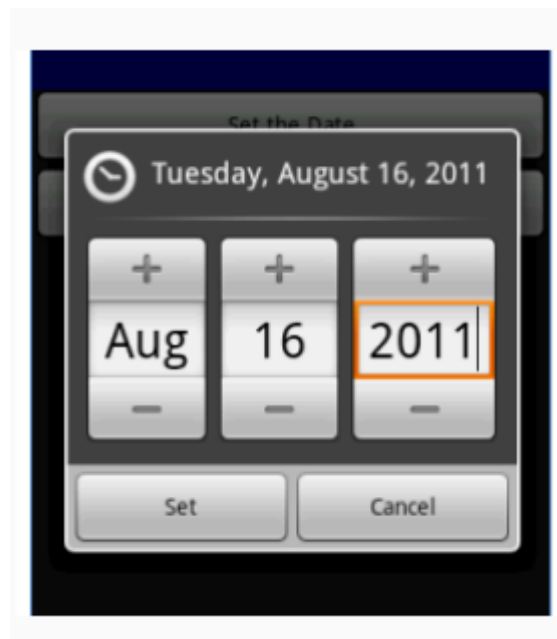
TimePickerDialog.OnTimeSetListener t = new TimePickerDialog.OnTimeSetListener()
{
    public void onTimeSet(TimePicker view, int hourOfDay, int minute) {
        myCalendar.set(Calendar.HOUR_OF_DAY, hourOfDay);
        myCalendar.set(Calendar.MINUTE, minute);
        updateLabel();
    }
};
```

```

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
    lblDateAndTime = (TextView) findViewById(R.id.lblDateAndTime);
    Button btnDate = (Button) findViewById(R.id.btnDate);
    btnDate.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            new DatePickerDialog(AndDemoUI.this, d,
                myCalendar.get(Calendar.YEAR),
                myCalendar.get(Calendar.MONTH),
                myCalendar.get(Calendar.DAY_OF_MONTH)).show();
        }
    });
    Button btnTime = (Button) findViewById(R.id.btnTime);
    btnTime.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            new TimePickerDialog(AndDemoUI.this, t,
                myCalendar.get(Calendar.HOUR_OF_DAY),
                myCalendar.get(Calendar.MINUTE), true).show();
        }
    });
    updateLabel();
}
}

```

- DatePicker & DatePickerDialog



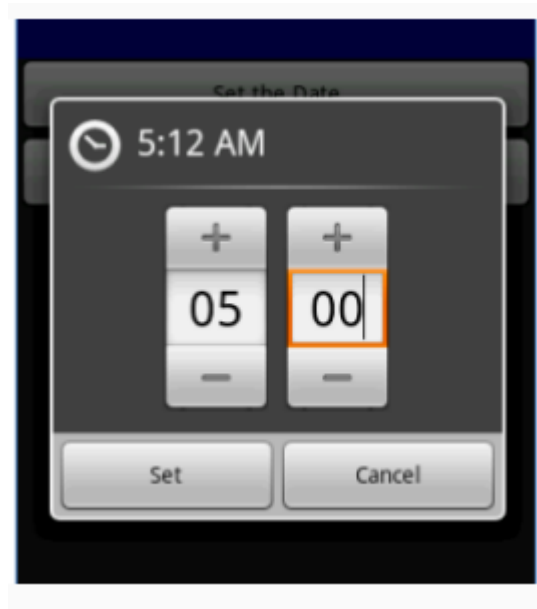
```

<Button
    android:id="@+id/btnDate"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Set the Date"
/>

```

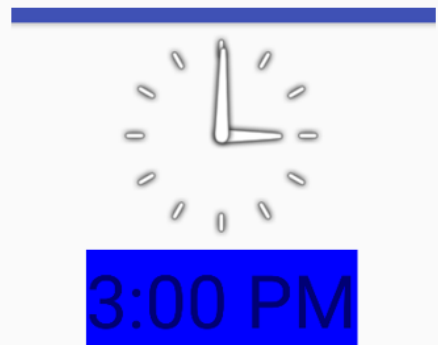
- month = 0 → 11

- return listener
(OnDateChangeListener/OnDateSetListener) when setting value
- TimePicker & TimePickerDialog

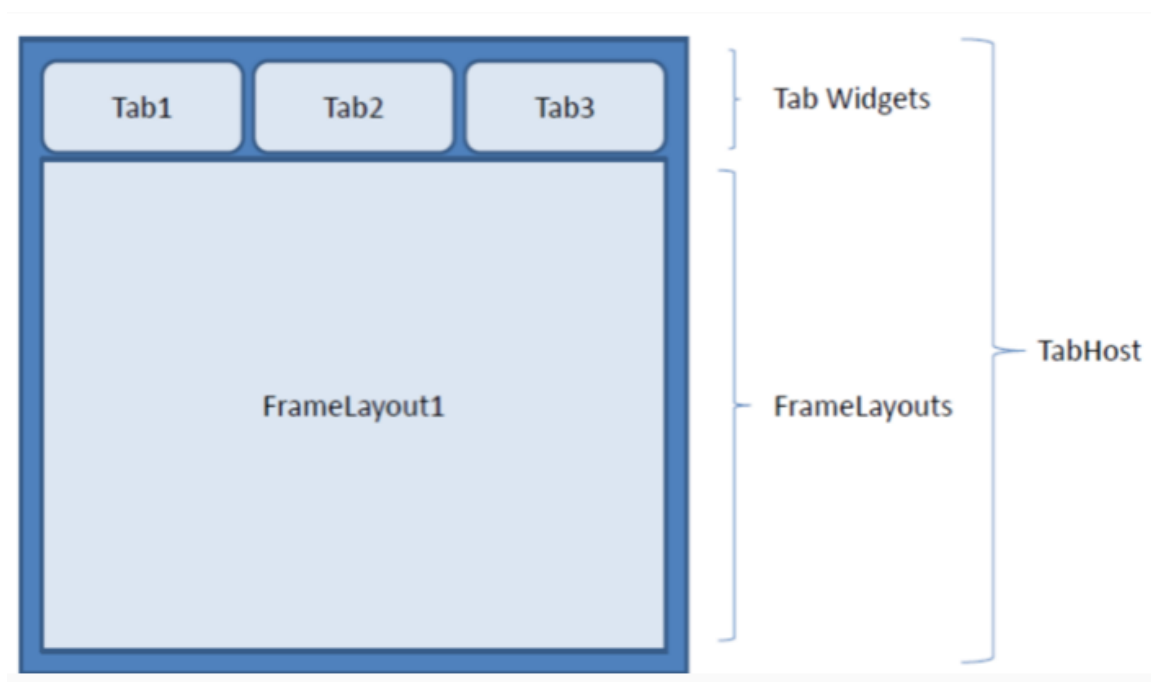


- set default time, user can change, 24-format
 - show format
 - return listener
(OnTimeChangeListener/OnTimeSetListener)
- Clock

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    android:id="@+id/widget34"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    xmlns:android="http://schemas.android.com/apk/res/android"
>
    <DigitalClock
        android:id="@+id/digital"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#ff0000ff"
        android:textSize="200px"
        android:layout_below="@+id/analog"
        android:layout_centerHorizontal="true"
    >
    </DigitalClock>
    <AnalogClock
        android:id="@+id/analog"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
    >
    </AnalogClock>
</RelativeLayout>
```



- Tab



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:orientation="vertical"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent">
  <TabHost android:id="@+id/tabhost"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <TabWidget android:id="@android:id/tabs"
      android:layout_width="fill_parent"
      android:layout_height="wrap_content"
    />
    <FrameLayout
      android:id="@android:id/tabcontent"
      android:layout_width="fill_parent"
      android:layout_height="fill_parent"
      android:paddingTop="62px">
      <!-- PUT HERE FrameLayout1 -->
      <!-- PUT HERE FrameLayout2 -->
    </FrameLayout>
  </TabHost>
</LinearLayout>
```

```

<!-- PUT HERE FrameLayout2 -->
<FrameLayout
    android:id="@+id/tab2"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <include
        layout="@layout/screen2" />
</FrameLayout>

```

```

TabHost tabs=(TabHost)findViewById(R.id.tabhost);
tabs.setup();
TabHost.TabSpec spec;
spec = tabs.newTabSpec("tag1");
spec.setContent(R.id.tab1);
spec.setIndicator("1-Clock");
tabs.addTab(spec);

spec=tabs.newTabSpec("tag2");
spec.setContent(R.id.tab2);
spec.setIndicator("2-Login");
tabs.addTab(spec);
tabs.setCurrentTab(0);

```

Set Tab1

Set Tab2

```

spec = tabs.newTabSpec("tag2");
spec.setContent(R.id.tab2);
spec.setIndicator("2-Login",
    getResources().getDrawable(R.drawable.ic_m
    enu_info_details));
tabs.addTab(spec);

```

