Android UI Control
[DatePicker]

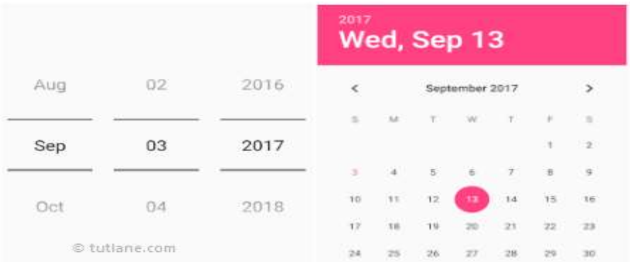
Android – UI Control

DatePicker

- **DatePicker** is a control that will allow users to select the date by a day, month and year in our application user interface.

- If we use **DatePicker** in our application, it will ensure that the users will select a valid date.

- Following is the pictorial representation of using a datepicker control in android applications.



- Generally, in android DatePicker available in two modes, one is to show the complete calendar and another one is to show the dates in spinner view.

Android – UI Control

Create Android DatePicker in XML Layout File

- We can create a DatePicker in XML layout file using **<DatePicker>** element with different attributes like as shown below

```
<DatePicker android:id="@+id/datePicker1"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content" />
```

- The DatePicker supports a two types of modes, those are **Calendar** and **Spinner** to show the date details in our application.

Android – UI Control

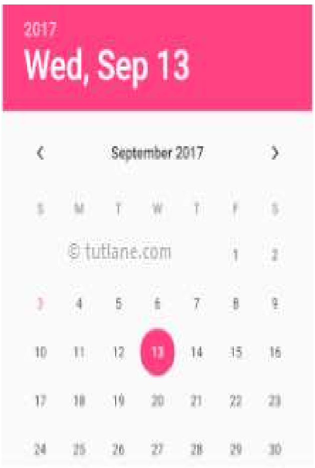
Android DatePicker with Calendar Mode

- We can define android DatePicker to show only a calendar view by using DatePicker android:datePickerMode attribute.

- Following is the example of showing the DatePicker in **Calendar** mode.

```
<DatePicker
   android:id="@+id/datePicker1"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:datePickerMode="calendar"/>
```

- The above code snippet will return the DatePicker in android like as shown in fig



- If you observe the above result we got the DatePicker in calendar mode to select a date based on our requirements.

Android – UI Control

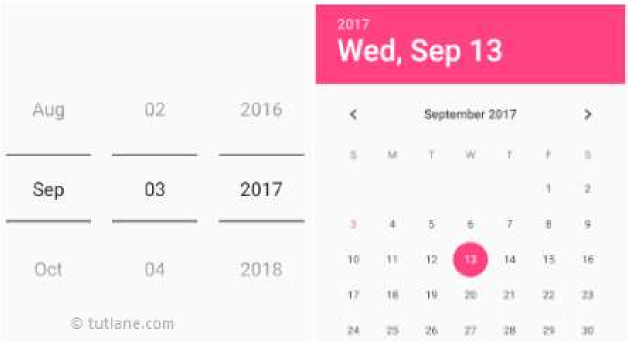
Android DatePicker with Spinner Mode

- If we want to show the DatePicker in spinner format like showing day, month and year separately to select the date, then by using DatePicker android:datePickerMode attribute we can achieve this.

- Following is the example of showing the DatePicker in Spinner mode.

```
<DatePicker
  android:id="@+id/datePicker1"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:datePickerMode="spinner"/>
```

- The above code snippet will return the DatePicker in android like as shown in fig



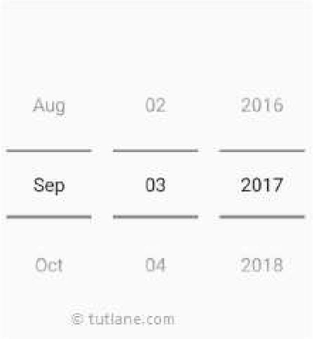
Android – UI Control

Android DatePicker with Spinner Mode

- If you observe the above result we got the DatePicker in both **Spinner** and **Calendar** modes to select the date.

- To get only spinner mode date selection, then we need to **set android:calendarViewShown="false"** attribute in DatePicker control like as shown below.

```
<DatePicker
   android:id="@+id/datePicker1"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:datePickerMode="spinner"
   android:calendarViewShown="false"/>
```



- The above code will return the DatePicker like as shown in fig

Android – UI Control

Android DatePicker Control Attributes

Attribute	Description
android:id	It is used to uniquely identify the control
android:datePickerMode	It is used to specify datepicker mode either spinner or calendar
android:background	It is used to set the background color for the date picker.
android:padding	It is used to set the padding for left, right, top or bottom of the date picker.

Android – UI Control

Android DatePicker Example

- In this example we define one **DatePicker** control, one **TextView** control and one **Button** control in **RelativeLayout** to show the selected date on **Button** click in the android application.

- Create a new android application using android studio and give names as **DatePickerExample**.

- Now open an **activity_main.xml** file from **\res\layout** path and write the code like as shown below

Android – UI Control

Android DatePicker Example

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk
/res/android"
   android:layout_width="match_parent"
android:layout_height="match_parent">
   <DatePicker
      android:id="@+id/datePicker1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_centerHorizontal="true"
      android:layout_marginTop="20dp" />
   <Button
      android:id="@+id/button1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_below="@+id/datePicker1"
      android:layout_marginLeft="100dp"
      android:text="Get Date" />
<TextView
      android:id="@+id/textView1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_below="@+id/button1"
      android:layout_marginLeft="100dp"
      android:layout_marginTop="10dp"
      android:textStyle="bold"
      android:textSize="18dp"/>
</RelativeLayout>
```

Android – UI Control

Android DatePicker Example

MainActivity.java

```
package com.tutlane.datepickerexample;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
   DatePicker picker;
   Button btnGet;
   TextView tvw;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      tvw=(TextView)findViewById(R.id.textView1);
      picker=(DatePicker)findViewById(R.id.datePicker1);
      btnGet=(Button)findViewById(R.id.button1);
      btnGet.setOnClickListener(new View.OnClickListener() {
```

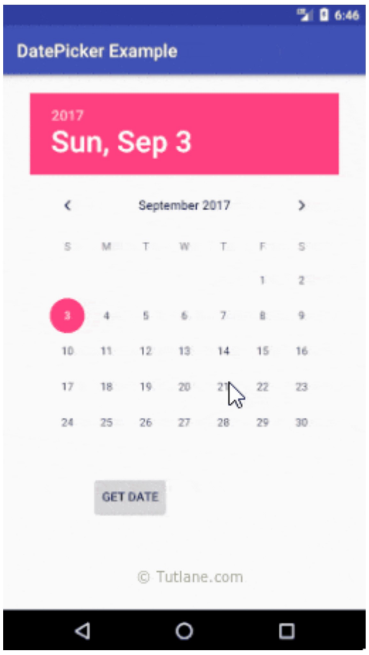
Android – UI Control

Android DatePicker Example

```
            @Override
            public void onClick(View v) {
                tvw.setText("Selected Date: "+ picker.getDayOfMonth()+"/"+
(picker.getMonth() + 1)+"/"+picker.getYear());
            }
        });
    }
}
```

Android – UI Control

Output of Android DatePicker Example



THANKS