2.3 Table, Grid, and Frame layouts

TableLayout

The TableLayout is a ViewGroup that arranges child Views in a table with rows and columns.

TableRow

Each table row is defined by a TableRow object, and each View in the TableRow forms a single table cell.

layout_span

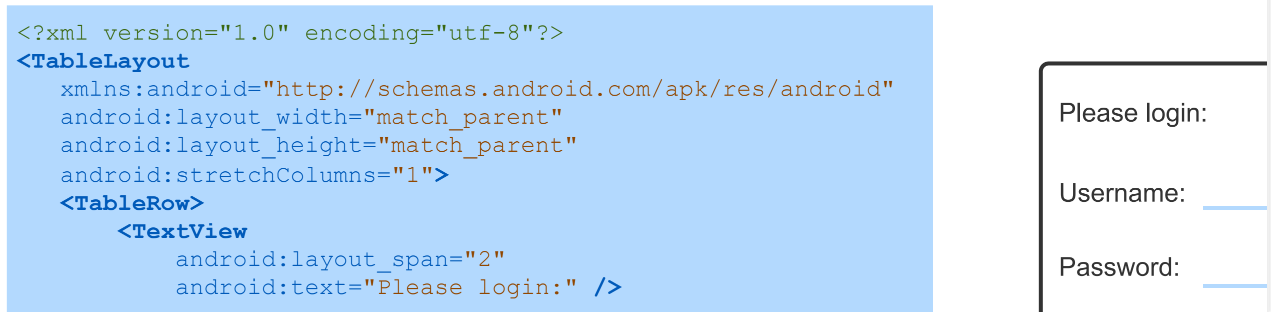
A cell may span more than one column when the View uses the layout_span parameter to indicate the number of columns to span.

layout_column

Cells may be left empty using the layout_column parameter to indicate the View's starting column, skipping columns to the left.

PARTICIPATION ACTIVITY 2.3.1: TableLayout example.

Start □ 2x speed

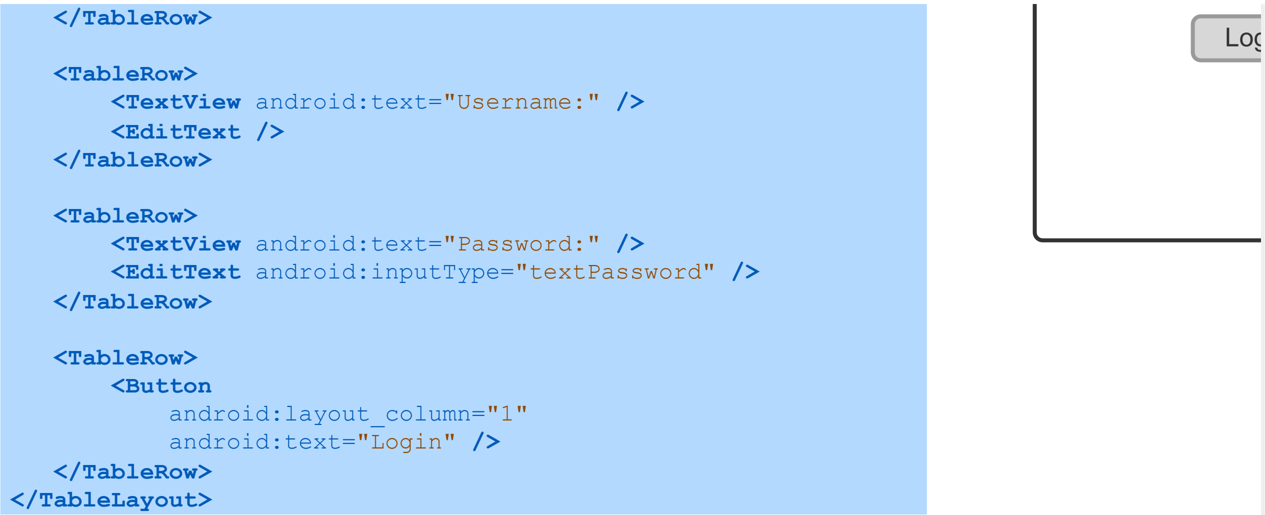


```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:stretchColumns="1">
   <TableRow>
       <TextView
           android:layout_span="2"
           android:text="Please login:" />
```

Please login:

Username:

Password:



```
    </TableRow>

    <TableRow>
        <TextView android:text="Username:" />
        <EditText />
    </TableRow>

    <TableRow>
        <TextView android:text="Password:" />
        <EditText android:inputType="textPassword" />
    </TableRow>

    <TableRow>
        <Button
            android:layout_column="1"
            android:text="Login" />
    </TableRow>
</TableLayout>
```

Captions

1. A TableLayout contains 4 TableRows that define the contents of each row.
2. Two Views exist in the second and third rows, so the table will have two columns.
3. The first row has a TextView that spans 2 columns.
4. The next two rows have a TextView in the first column and an EditText in the second column.
5. The last row has a single Button that begins in column 1.

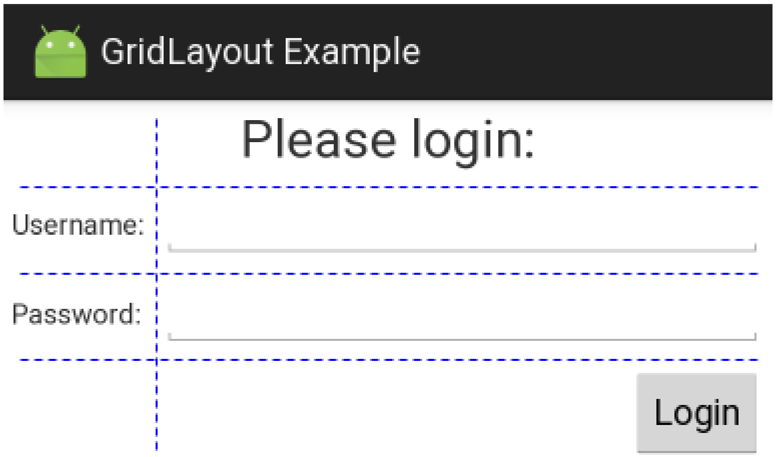
Feedback?

GridLayout

The GridLayout is a ViewGroup that arranges child Views in a grid. By default, each child View occupies a single grid cell.

Figure 2.3.1: GridLayout example with two columns.

```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:useDefaultMargins="true"
    android:columnCount="2">
    <TextView
        android:textSize="26sp"
        android:layout_columnSpan="2"
        android:layout_gravity="center_horizontal"
        android:text="Please login:" />
    <TextView
        android:text="Username:" />
    <EditText
        android:layout_gravity="fill_horizontal" />
    <TextView
        android:text="Password:" />
    <EditText
        android:layout_gravity="fill_horizontal"
        android:inputType="textPassword" />
    <Button
        android:layout_column="1"
        android:layout_gravity="right"
        android:text="Login" />
</GridLayout>
```



Feedback?

FrameLayout

A FrameLayout is a ViewGroup that displays a single View or overlays multiple Views.

layout_gravity

The layout_gravity parameter aligns the View within the layout using constants like `center`, `left`, `right`, `top`, `bottom`, etc.

gravity

The gravity parameter aligns the content within the View using values like `center`, `left`, `right`, `top`, `bottom`, etc.

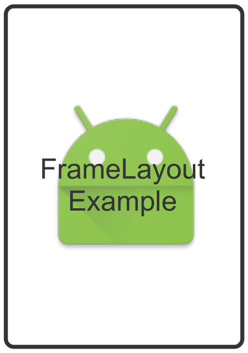
PARTICIPATION ACTIVITY 2.3.4: FrameLayout example.

Start □ 2x speed

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">

    <ImageView
        android:src="@mipmap/ic_launcher"
        android:layout_gravity="center"
        android:layout_height="300dp"
        android:layout_width="300dp" />

    <TextView
        android:text="FrameLayout Example"
        android:textSize="50sp"
        android:layout_height="fill_parent"
        android:layout_width="fill_parent"
        android:gravity="center" />
</FrameLayout>
```



Captions

1. A FrameLayout uses layout_width and layout_height to fill the entire screen.
2. The ImageView displays the ic_launcher image resource centered in the layout using layout_gravity.
3. The TextView fills the entire layout, and gravity centers the text content in the TextView, so the text appears on top of the ImageView.

Feedback?

This section does not contain presentation elements.

How was this section? 👍 👎 **Provide feedback**