

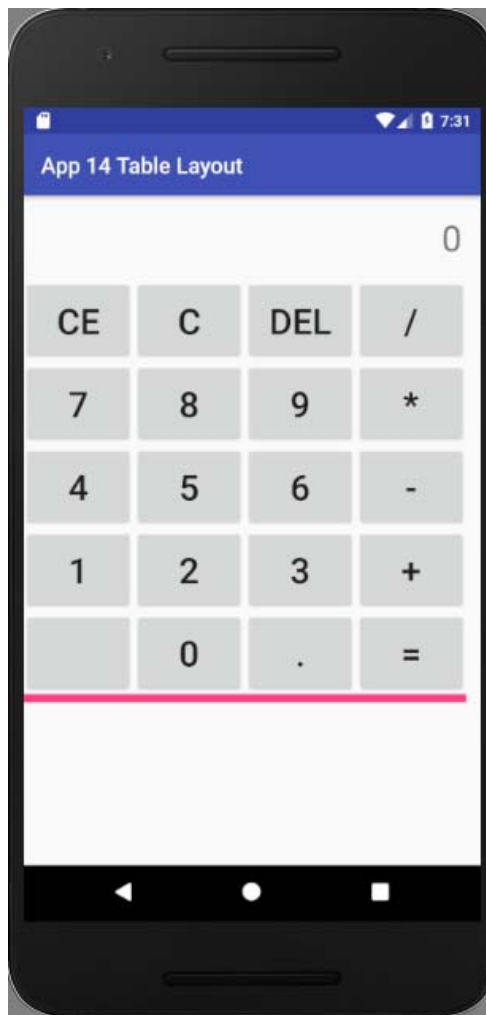
PRACTICA 14 TableLayout.

El alumno aprenderá a usar el TableLayout creando como aplicación una calculadora que se ajusta a este tipo de aplicaciones tablares.

OBJETIVOS ESPECÍFICOS:

- Utilizar TableLayout como contenedor principal de la aplicación.
- Crear objetos Actividad:
 - Crear 1 TextView como pantalla de la calculadora.
 - Crear 20 botones que simulen las teclas de la calculadora.

NOTA: Por el momento solo será crear el front-end de la aplicación y la lógica quedara a futuras practicas.



Este será nuestro front-end final.

Basados en la practica anterior, en el uso de los recursos de Android, aplicaremos estos conocimientos para desarrollar esta aplicación, por lo tanto, accederemos a la carpeta values y dentro de ella al archivo strings.xml y agregaremos el siguiente código:

```
<string name="btnUno">1</string>
<string name="btnDos">2</string>
<string name="btnTres">3</string>
<string name="btnCuatro">4</string>
<string name="btnCinco">5</string>
<string name="btnSeis">6</string>
<string name="btnSiete">7</string>
<string name="btnOcho">8</string>
<string name="btnNueve">9</string>
<string name="btnCero">0</string>
<string name="btnPunto">.</string>
<string name="btnMas">+</string>
<string name="btnMenos">-</string>
<string name="btnMultiplicar">*</string>
<string name="btnDividir">/</string>
<string name="btnCE">CE</string>
<string name="btnC">C</string>
<string name="btnBorrar">Del</string>
<string name="btnIgual">=</string>
```

Cada uno de ellos es el texto que desplegará cada botón.

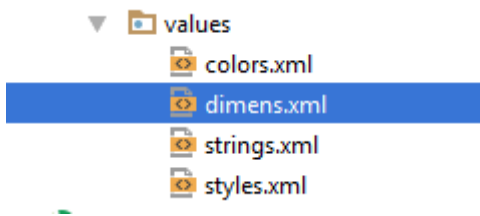
Posterior a esto, accedemos a la carpeta styles.xml para definir los estilos de nuestros objetos gráficos de nuestra aplicación, agregue el siguiente código:

```
<style name="dccBoton">
    <item name="android:layout_width">match_parent</item>
    <item name="android:layout_height">wrap_content</item>
    <item name="android:gravity">center</item>
    <item name="android:textSize">@dimen/space32</item>
    <item name="android:padding">@dimen/space16</item>
</style>

<style name="dccTexto">
    <item name="android:layout_width">match_parent</item>
    <item name="android:layout_height">wrap_content</item>
    <item name="android:textSize">@dimen/space32</item>
    <item name="android:padding">@dimen/space16</item>
    <item name="android:gravity">right</item>
</style>
```

Android le indicara que existen errores en algunos atributos definidos, no se preocupe, vamos a resolverlo.

Vamos a crear un archivo de nombre `dimens` en la carpeta `values`:



Agregaremos el siguiente código:

```
<dimen name="space02">02dp</dimen>
<dimen name="space04">04dp</dimen>
<dimen name="space08">08dp</dimen>
<dimen name="space16">16dp</dimen>
<dimen name="space32">32dp</dimen>
<dimen name="space64">64dp</dimen>
```

Con este archivo de dimensiones se corrigen los errores del archivo `styles.xml`.

NOTA: Esto es una práctica de programación para que a futuro el mantenimiento sea mas simple y lleve menos tiempo y no teniendo efectos secundarios al no modificar la actividad o bien la clase java que hace referencia a la actividad.

Una vez hecho esto, accedemos a la actividad principal y procedemos a crear cada uno de los elementos del front-end.

En primera instancia creamos el `TableLayout`.

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout

    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:stretchColumns="*">

    *** Elementos graficos.

</TableLayout>
```

NOTA: Si tiene otro layout, sustitúyalo como se indica

Ahora dentro de este `TableLayout` el atributo `stretchColumns` tiene la función de distribuir el contenido de cada `TableRow` en función de la fila que tenga mas elementos.

Dentro del TableLayout, agregaremos cada uno de los elementos como se indica a continuación:

```
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:stretchColumns="*"
    tools:context="com.example.delio.clase14tablelayout.MainActivity">

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        >
        <TextView
            android:text="0"
            android:layout_span="5" // Hace una celda que abarca las 5 columnas
            style="@style/dccTexto" // Aplica el estilo definido en style
        />

    </TableRow>

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        >
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/btnCE"
            style="@style/dccBoton" // Define el estilo de los botones
        />
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/btnC"
            style="@style/dccBoton"
        />
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/btnBorrar"
            style="@style/dccBoton"
        />
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/btnDividir"
            style="@style/dccBoton"
        />
    </TableRow>

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        >
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/btnSiete"
```

```

        style="@style/dccBoton"
    />
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/btnOcho"
    style="@style/dccBoton"
/>
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/btnNueve"
    style="@style/dccBoton"
/>
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/btnMultiplicar"
    style="@style/dccBoton"
/>
</TableRow>

<TableRow
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnCuatro"
        style="@style/dccBoton"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnCinco"
        style="@style/dccBoton"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnSeis"
        style="@style/dccBoton"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnMenos"
        style="@style/dccBoton"
    />
</TableRow>

<TableRow
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/dccBoton"
        android:text="@string/btnUno"
    />
    <Button

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/dccBoton"
        android:text="@string/btnDos"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnTres"
        style="@style/dccBoton"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnMas"
        style="@style/dccBoton"
    />
</TableRow>

<TableRow
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/dccBoton"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnCero"
        style="@style/dccBoton"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnPunto"
        style="@style/dccBoton"
    />
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/btnIgual"
        style="@style/dccBoton"
    />
</TableRow>

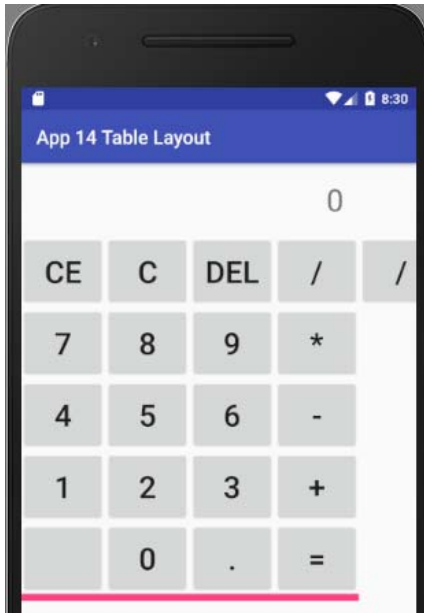
<TableRow
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >
    <TextView
        android:layout_width="match_parent"
        android:layout_height="@dimen/space08"
        android:background="@color/colorAccent"
        android:layout_span="4"
        android:text=""
    />
</TableRow>
</TableLayout>

```

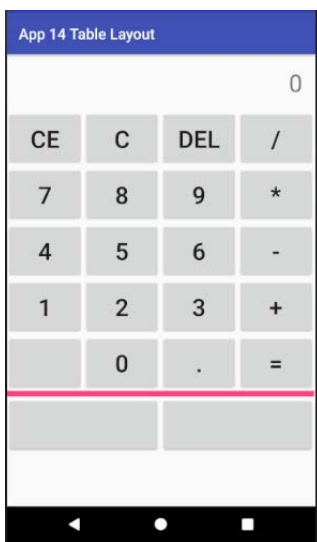
Con estos pasos, ya creamos el front-end de nuestra aplicación,

NOTAS: El table layout está en función de sus TableRows, cada TableRow almacena una serie de elementos que se acomodan en forma secuencial en función de su posición dentro del TableRow.

Si una TableRow tiene 6 elementos, entonces será una cuadrícula de seis filas.



También podemos hacer distribuciones como se ilustra en la última TableRow, donde dos botones ocupan dos columnas cada uno mediante el uso del `android:layout_span="2"`



Bien, la lógica le toca a usted implementarla jejeje