

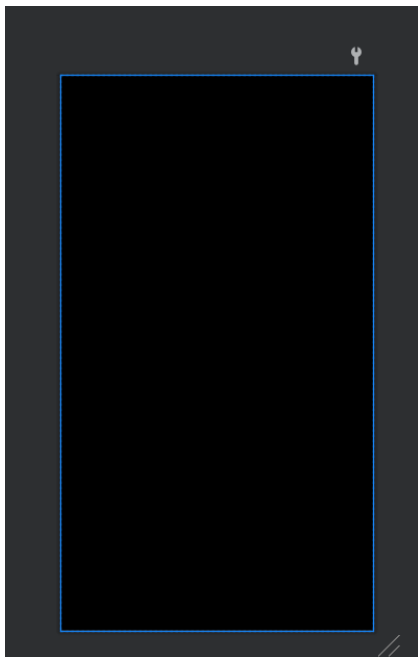
Calculadora

Interfaz:

Lo primero que he querido hacer es establecer el color de fondo en negro para dejarlo más elegante.

```
android:background="@color/black"
```

Quedará así:



Lo siguiente es crear líneas para así ser más fácil la colocación de botones.

Se hace con Guideline

```
<androidx.constraintlayout.widget.Guideline
    android:id="@+id/primerVertical"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    app:layout_constraintGuide_begin="36dp"
    app:layout_constraintGuide_percent="0.01"/>
```

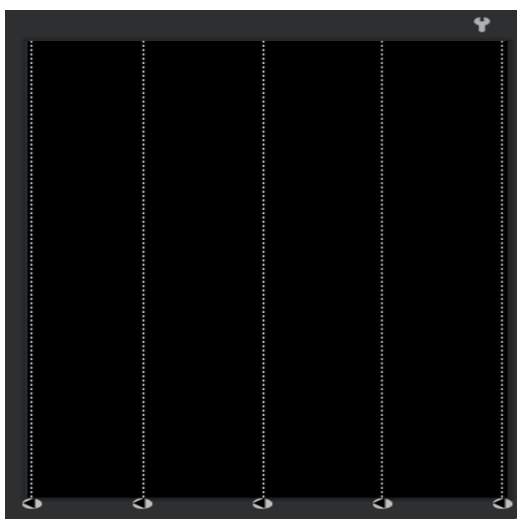
Esta es la primera línea



Lo siguiente es crear todas las líneas verticales siguiendo el patrón de `id+porcentaje(Guide_percent)`

```
<androidx.constraintlayout.widget.Guideline
    android:id="@+id/segundaVertical"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    app:layout_constraintGuide_begin="36dp"
    app:layout_constraintGuide_percent="0.24"/>

<androidx.constraintlayout.widget.Guideline
    android:id="@+id/terceraVertical"
    android:layout_width="wrap_content"
```



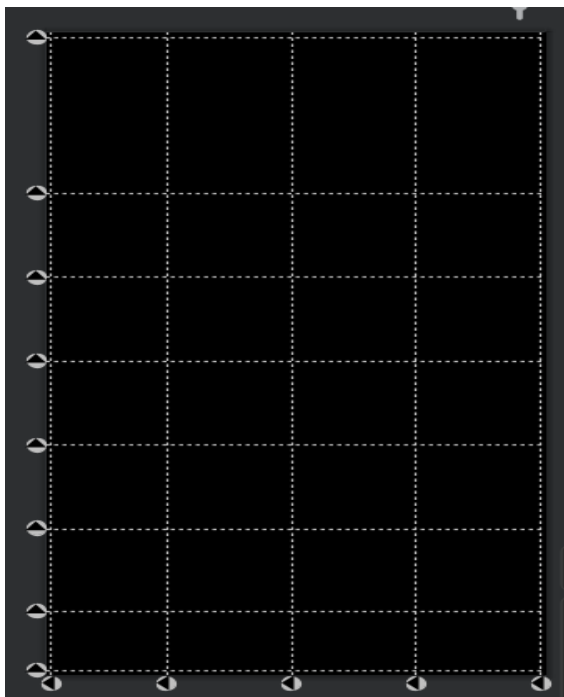
Tras hacer las verticales hacemos las horizontales. Lo único que hay que cambiar es la propiedad android:orientation de vertical a horizontal.

```
<androidx.constraintlayout.widget.Guideline
    android:id="@+id/primeraHorizontal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_begin="36dp"
    app:layout_constraintGuide_percent="0.01"/>

<androidx.constraintlayout.widget.Guideline
    android:id="@+id/segundaHorizontal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_begin="258dp"
    app:layout_constraintGuide_percent="0.25" />

<androidx.constraintlayout.widget.Guideline
```

Este es el resultado:



A partir de aquí es crear objetos que estén limitados dentro de las líneas.

El primero, es el resultado/interfaz de la calculadora. Para ello utilicé un textView.

```
<TextView
    android:id="@+id/resultado"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_margin="10dp"
    android:background="#FF9800"
    android:gravity="right|center_vertical"
    android:text="0"
    android:textColor="@color/black"
    android:textSize="35dp"
    app:layout_constraintBottom_toBottomOf="@+id/segundaHorizontal"
    app:layout_constraintEnd_toEndOf="@id/ultimaVertical"
    app:layout_constraintStart_toStartOf="@+id/primerVertical"
    app:layout_constraintTop_toTopOf="@+id/primerHorizontal" />
```

Al usar 0dp en width y height, el textView ocupa todo(dentro de sus limitaciones), utilizo el layout_margin para alejarlo un poco del cuadro.

Background es el color, gravity es para que flote a la derecha y al centro de forma vertical.

Las constraints son muy necesarias para la limitación. Para ello anteriormente creé las líneas. En este caso, mi resultado ocupa mucha pantalla.



Ahora a colocar los botones.

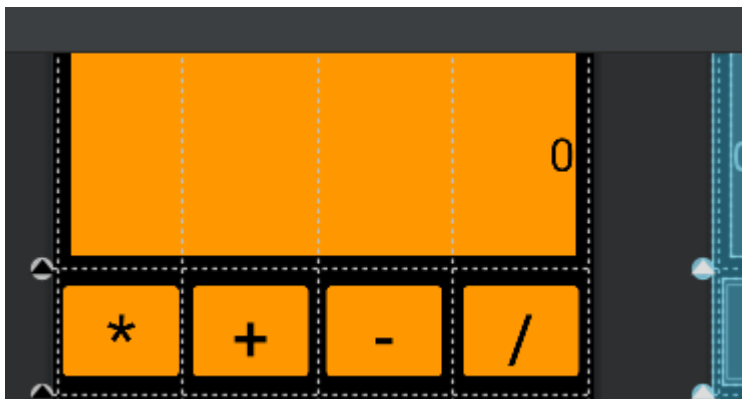
Todos siguen el mismo patrón:

```

<Button
    android:id="@+id/borrar"
    android:layout_margin="5dp"
    android:layout_width="wrap_content"
    android:layout_height="80dp"
    android:backgroundTint="#FF9800"
    android:text="C"
    android:textSize="50dp"
    android:textColor="@color/black"
    app:layout_constraintBottom_toTopOf="@+id/septimaHorizontal"
    app:layout_constraintEnd_toStartOf="@+id/ultimaVertical"
    app:layout_constraintStart_toStartOf="@+id/cuartaVertical"
    app:layout_constraintTop_toTopOf="@+id/sextaHorizontal" />

```

Con las líneas separando por cuadrados y el wrap_content del layout_width logro que cada botón ocupe más o menos todo su cuadrado.



Lo siguiente es muy parecido, dar un ID establecer el tamaño, este es el resultado final.

