# Row and Column in Compose

## Row:

setContent{

Row{

Text(“hello”)

Text{“Big bang”)

}

}

## Column:

Arrangement -> Main Axis column k direction hi

Arrangement.SpaceAround, Arrangement.SpaceBetween, Arrangement.SpaceEvenly

Alignment -> Cross Axis , column ko cross krny wali direction mai

setContent{

Column {

Text(“hello”)

Text{“Big bang”)

}

}

Column (modifier = Modifier.*fillMaxHeight*().*background*(Color.Red),  
 horizontalAlignment = Alignment.Start,

verticalArrangement = Arrangement.SpaceAround

) **{** Text("Big N")  
 Text("Big O")  
**}**

# Fill Max Size

Modifier.*fillMaxSize(0.5f) -> It fills 50% of screen*

# Width and Height

modifier = Modifier

.width(100.dp),

.height(100.dp)

# Modifier Attributes on Column/Row

.background(Color.Red)

.fillMaxHeight(), .fillMaxWidth(), .width(), .height(), .fillMaxSize()

You can pass paremeters to Maximum height, width and fill maxsize like 0.5f 50% covering.

**If we pass more width than parents width, automatically set to parent width for width() function**

**.requiredWidth will actually taken according to given width, so it does not rely on parents width.**

.padding(10.dp) You can give padding like that.

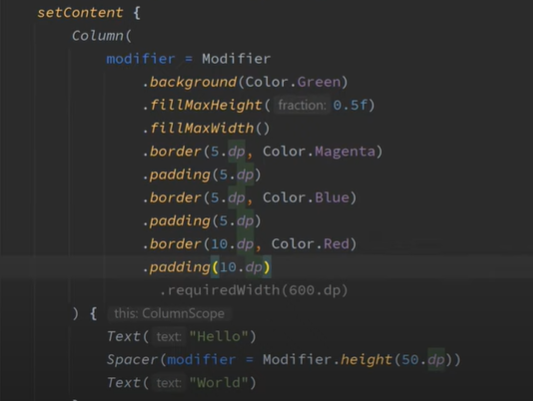
**2 paremeters -> first for horinzontal and second for vertical**

**4 parameters -> start, top, end, bottom**

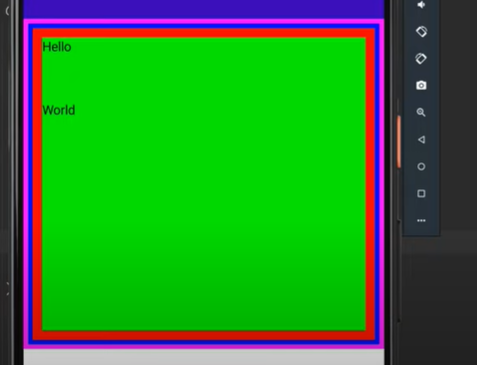
**Note:**

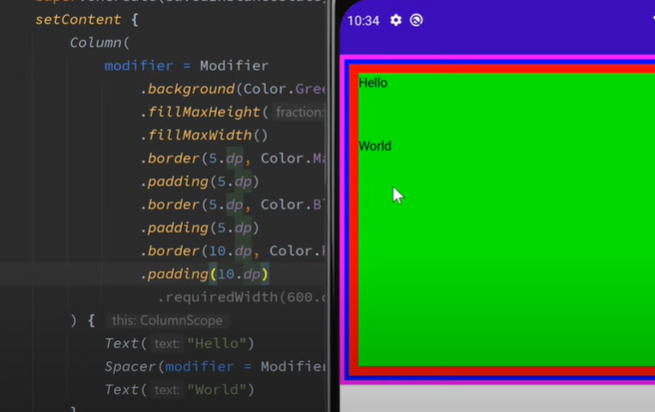
In jetpack, we use paddings for margin

.border (5.dp, Color.Red)



Output:





# 

# Modifier Attributes on Text

Text(“Talha”, modifier = Modifier.offset(50.dp, 20.dp)

50.dp -> right margin

20.dp -> bottom

It does not push other elements, it only takes given elements.

modifier = Modifier.clickable {}

# Spacer

We can use this to separate UI items and use modifier inside it.

Spacer(modifier = Modifier.height(50.dp))

Text(“World”)