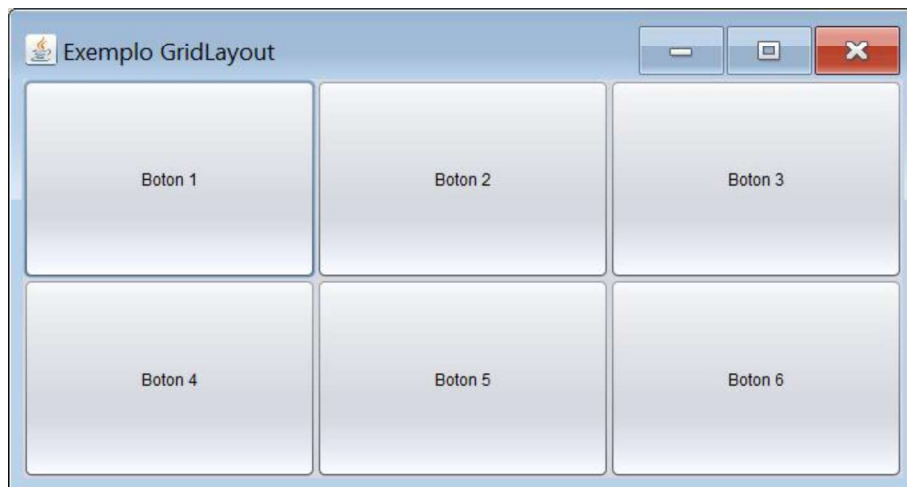


#### 1.1.1.1 Layout manager GridLayout

O layout manager GridLayout defínese mediante a clase `java.awt.GridLayout`. O seu aspecto visual é o seguinte:



Layout manager GridLayout aplicado con despregue de compoñentes nunha soa fila.



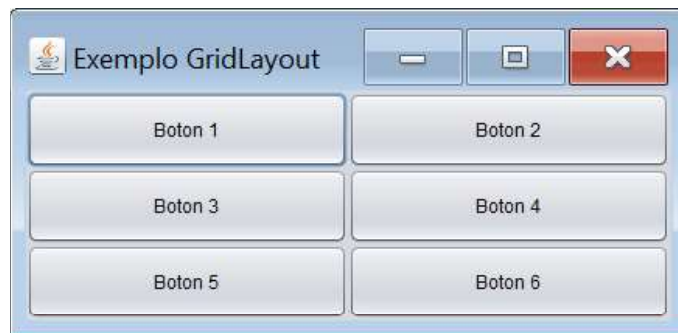
Layout manager GridLayout aplicado con despregue de compoñentes en dous filas e tres columnas.



Layout manager GridLayout aplicado con despregue de compoñentes en tres filas e dous columnas.

Cando aplicamos un layout manager GridLayout sobre un contedor, o espazo para situar compoñentes dentro del configúrase como unha matriz de celas na cal cada unha das celas ten o mesmo tamaño. Cada compoñente que engadimos ao contedor sitúase dentro dunha das celas e ocupa o tamaño desa cela. O número de celas presentes no contedor podemos configuralo indicando o número de filas e o número de columnas que desexamos que teña a matriz de celas.

Na seguinte imaxe podemos ver o que ocorre no caso de que redimensionemos o formulario sobre o que aplicamos o layout manager GridLayout (redimensionamento aplicado sobre a última imaxe vista anteriormente):



Como pódese observar, o layout manager GridLayout conserva a configuración respecto ao número de filas e de columnas establecidas para a súa matriz de celas. Ao modificarse o espazo dispoñible no contedor, o que ocorre é que este espazo redistribúese dun xeito equitativo entre as celas. Os compoñentes situados nas celas empregan todo o espazo dispoñible dentro de cada cela.

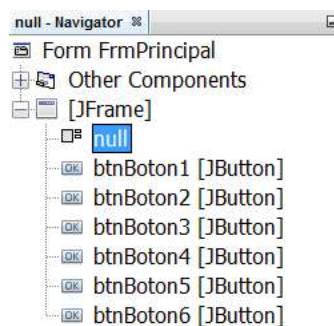
Ademais de configurar o número de filas e de columnas que compoñen a matriz de celas do layout manager GridLayout, tamén é posible configurar a separación vertical e horizontal entre cada cela da matriz.

## Configuración dun GridLayout empregando NetBeans

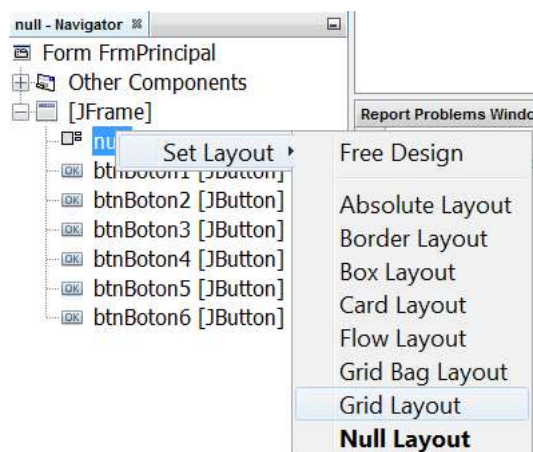
A continuación imos desenvolver o seguinte formulario:



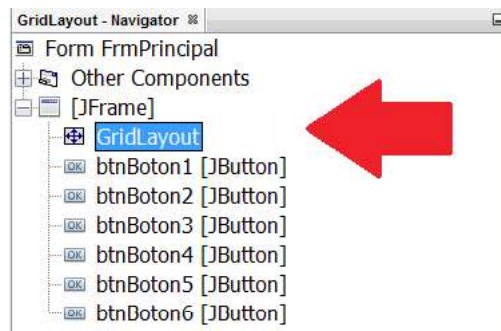
Para comenzar, creamos un proxecto e engadimos nel un formulario. Dentro do noso formulario engadimos seis botóns.



O seguinte que debemos facer é indicar que queremos establecer un layout manager de tipo GridLayout sobre o noso formulario. Para elo prememos sobre JFrame co botón dereito, eliximos a opción Set Layout e seleccionamos Grid Layout:



A partir deste momento temos establecido para o noso formulario como layout manager un layout manager de tipo GridLayout:

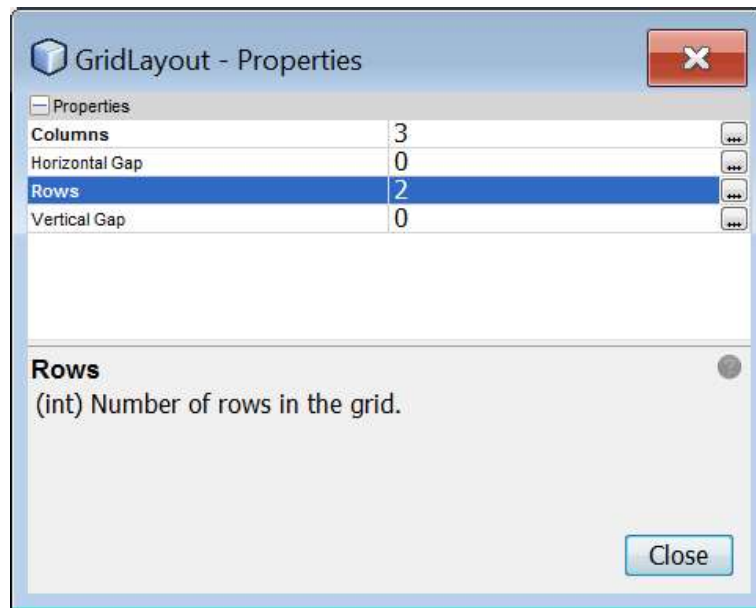


Agora xa temos configurado o noso contedor cun GridLayout e se executamos a aplicación desenvolvida este é o resultado:



Como pódese observar na imaxe anterior, non logramos exactamente o obxectivo perseguido. Cando establecemos un layout manager GridLayout, por defecto este configúrase para colocar tódolos compoñentes dentro do contedor nunha soa fila.

Para modificar o aspecto da malla de celas do layout manager GridLayout debemos modificar as súas propiedades. Para elo accedemos ás propiedades do layout manager GridLayout e modificamos os valores das propiedades Columns e Rows.

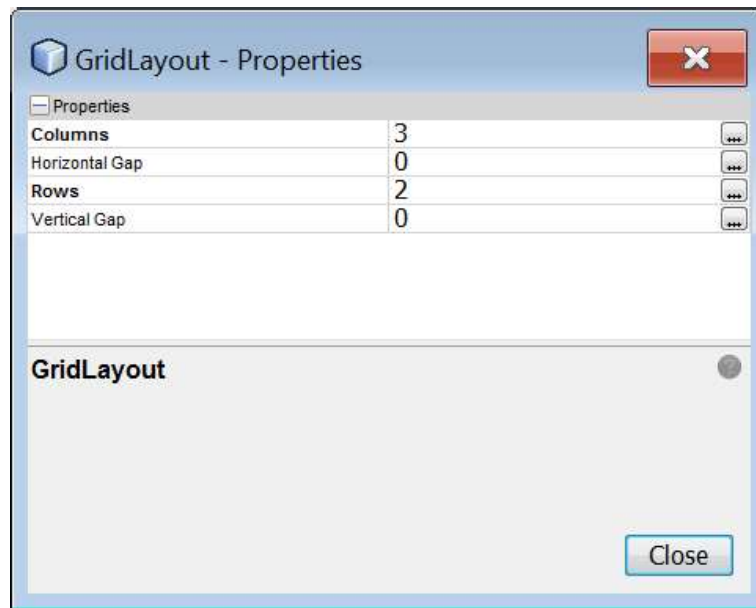


Coa propiedade Columns indicamos cantas columnas queremos que compoñan a nosa malla de celas. Coa propiedade Rows indicamos cantas filas queremos que compoñan a nosa malla de celas. Tras modificar as propiedades indicadas anteriormente, este é o resultado de executar a nosa aplicación:

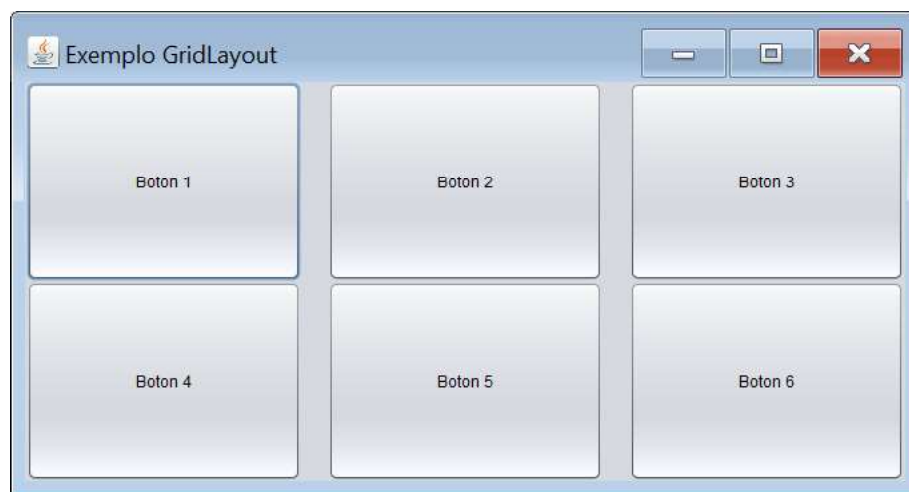


Agora sí acadamos o aspecto que buscábamos para a distribución de compoñentes do noso layout manager GridLayout.

Se queremos modificar a separación horizontal e/ou vertical entre as celas da matriz do layout manager GridLayout, debemos modificar as propiedades do layout manager GridLayout denominadas Horizontal Gap e Vertical Gap:



Mediante a propiedade Horizontal Gap establecemos a separación horizontal entre as celas da matriz do layout manager GridLayout. Na seguinte imaxe pódese observar que ocorre se establecemos a propiedade HorizontalGap a 20 píxels:



Na seguinte imaxe pódese observar que ocorre se establecemos a propiedade VerticalGap a 20 píxels:



Na seguinte imaxe pódese observar que ocorre se establecemos a propiedade `HorizontalGap` a 90 píxels e a propiedade `VerticalGap` a 20 píxels:



É importante recalcar que o espazo establecido como separación entre os compoñentes mantense fixo mentres que quen gaña ou perde espazo en función da redimensión do contedor é a propia cela da matriz; p.e., se redimensionamos o formulario obtemos o seguinte resultado:



Como pódese observar, o espazo de separación horizontal e vertical entre as celas da matriz do layout manager `GridLayout` mantense constante. O espazo

perdido polo formulario ao redimensionarse é collido das celas do layout manager GridLayout, co cal os botóns teñen menos espazo para debuxarse