

### TP n° 3

Exercice 1 : On veut tester l'afficheur `GridBagLayout`. Vous trouverez (en correction à la fin du sujet) le programme correspondant à l'affichage des 7 boutons de la fenêtre ci-dessous. Pour gagner du temps, vous pouvez chercher simplement ici à programmer le positionnement des 6 boutons n° 1, 2, 3, 5, 6 et 7 (i.e. vous ignorez le bouton 4 sur la 3<sup>ème</sup> ligne de la figure – et placerez ceux des lignes 1, 2 et 4).



Disposition : Le premier bouton sera agrandi verticalement pour avoir une hauteur de 40 pixels (cf. l'attribut `ipady`). Sur la dernière rangée, les boutons occupent chacun une cellule et remplissent uniformément l'espace en largeur en cas de changement de taille (leur affecter un même poids, par exemple 0.5 en `weightx`).

Note sur le sujet complet : le bouton 4 est positionné en bas de sa cellule (cf. l'attribut `anchor`) et laisse un espace de 10 pixels au dessus de lui (attribut `insets`). On peut aussi faire en sorte qu'il absorbe tout l'espace vertical disponible quand on agrandit la fenêtre (avec un `weighty` de 1.0).

Indications de correction :

```
import java.awt.*;  
import java.awt.event.*;  
import javax.swing.*;  
  
public class GridBagWindow extends JFrame {  
    boolean inAnApplet = true;  
    final boolean shouldFill = true;
```

```

final boolean shouldWeightX = true;

public GridBagWindow () {
    JButton button;
    Container contentPane = getContentPane();
    GridBagLayout gridbag = new GridBagLayout();
    GridBagConstraints c = new GridBagConstraints();
    contentPane.setLayout(gridbag);
    if (shouldFill) {
        // hauteur naturelle, largeur maximum
        c.fill = GridBagConstraints.HORIZONTAL;
    }

    button = new JButton("Bouton 1");
    c.ipady = 40; // hauteur de 40 s'il est plus petit
    c.gridx = 0;
    c.gridy = 0;
    c.gridwidth = 3; // toute la largeur car 3
colonnes
    gridbag.setConstraints(button, c);
    contentPane.add(button);

    button = new JButton("2");
    c.ipady = 0; // remise a zero
    c.gridwidth = 1; // remise au default
    c.gridx = 0;
    c.gridy = 1;
    gridbag.setConstraints(button, c);
    contentPane.add(button);

    button = new JButton("3");
    c.gridx = 2;
    c.gridy = 1;
    gridbag.setConstraints(button, c);
    contentPane.add(button);

    button = new JButton("Bouton 4");
    c.weighty = 1.0 ; // tout l'espace vert. suppl
    c.gridwidth = 2; // occupe deux colonnes
    c.gridx = 1;
    c.gridy = 2;
    c.insets = new Insets(10,0,0,0); //top padding
    c.anchor = GridBagConstraints.SOUTH; // en bas
    gridbag.setConstraints(button, c);
    contentPane.add(button);

    if (shouldWeightX) {
        c.weightx = 0.5; // le meme pour les 3 boutons
        // => meme taille
    }
    button = new JButton("5");
    c.gridx = 0;

```

```

c.gridy = 3;
c.gridwidth = 1; // occupe une seule colonne
c.insets = new Insets(0,0,0,0); //pas de padding
gridbag.setConstraints(button, c);
contentPane.add(button);

button = new JButton("Bouton 6");
c.gridx = 1;
c.gridy = 3;
gridbag.setConstraints(button, c);
contentPane.add(button);

button = new JButton("7");
c.gridx = 2;
c.gridy = 3;
gridbag.setConstraints(button, c);
contentPane.add(button);

addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent e) {
        if (inAnApplet) {
            dispose();
        } else {
            System.exit(0);
        }
    }
});
}

public static void main(String args[]) {
    GridBagWindow window = new GridBagWindow();
    window.inAnApplet = false;

    window.setTitle("GridBagLayout");
    window.pack();
    window.setVisible(true);
}
}

```