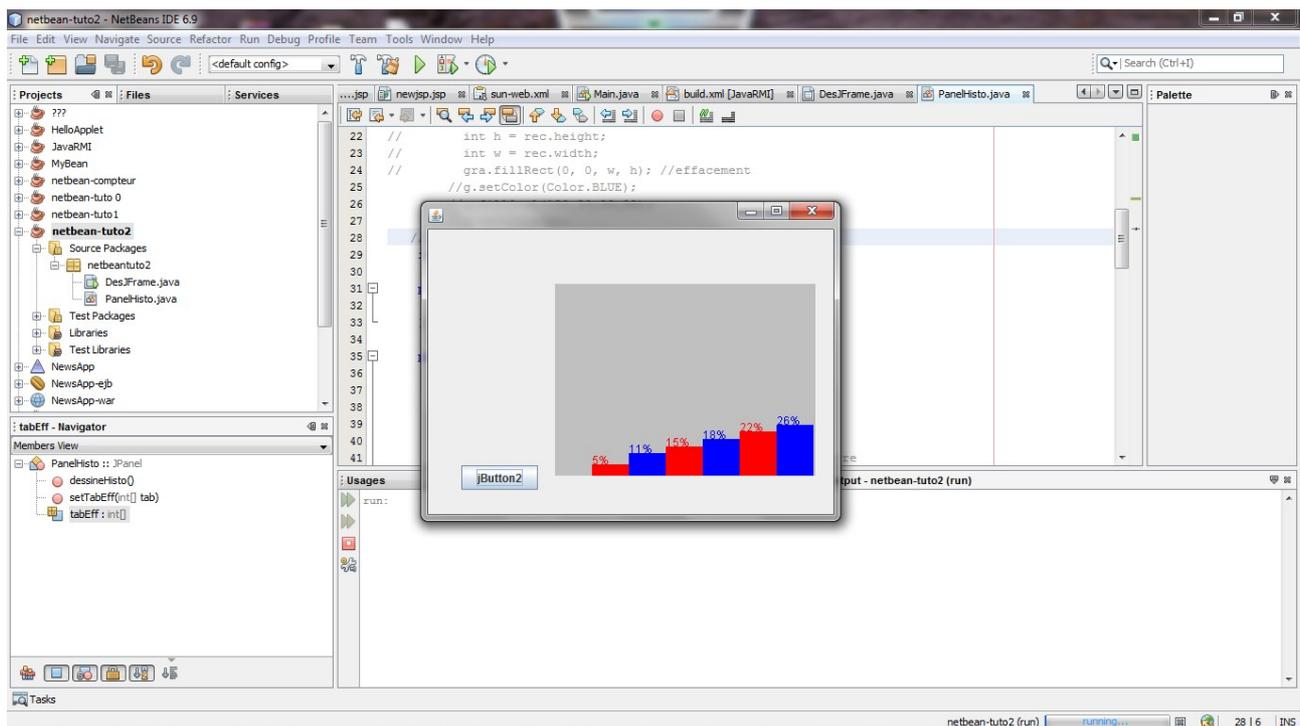


## Netbean tuto 2

L'objectif de ce tutorial est de vous guider dans l'affichage d'un histogramme lorsqu'on clique sur un bouton. Nous voulons dessiner un histogramme des **fréquences** qui se met à jour au fur et à mesure des tirages d'un dé. Il doit y avoir autant de **barres** que de **faces** du dé tiré. La hauteur de chaque barre doit être proportionnelle à la fréquence d'apparition de la face.

Cependant, dans cet exemple, on ne fera pas le tirage des dés mais juste l'affichage d'un histogramme à partir d'un tableau (tabEff) qui contiendra les informations à afficher. Dans ce tuto, tabEff aura une taille fixe correspondant à un dé à 6 faces. **tabEff[0]** contient l'effectif total. **tabEff[i]** contient le nombre d'apparition de la face i. Ici le tableau est donc de taille 7.

Voici une capture de l'exemple :



Etape 1 : Creation du projet : netbeantuto-2

Etape 2 : Création d'une JFrame : DesJFrame

Etape 3 : Creation d'une sous classe de JPanel : PanelHisto

3.a : créer une nouvelle classe Java PanelHisto

3.b : Ajouter un champ pour tabEff : `int[] tabEff = {53, 3, 6, 8, 10, 12, 14}; // valeurs par default du tableau.`

3.c : Ajouter un setter sur le tableau

```
public void setTabEff(int[] tab) {  
    this.tabEff = tab;  
}
```

3.d : Ajouter la fonction dessinerHisto qui sera appelée lorsqu'on clique sur le bouton. Essayer de comprendre le fonctionnement de cette méthode.

```
public void dessineHisto() {  
    double eff;  
    Rectangle rec = getBounds();  
    Graphics gra = getGraphics();  
    int h = rec.height;  
    int w = rec.width;
```

```

int la = (int) w / tabEff.length;//largeur d'une barre
gra.setColor(Color.lightGray);
gra.fillRect(0, 0, w, h); //effacement
if (tabEff[0] != 0) {
    for (int i = 1; i < tabEff.length; i++) {
        if (i % 2 == 1) {
            gra.setColor(Color.red); //couleur pour i impair
        } else {
            gra.setColor(Color.blue); //couleur pour i pair
        }
        eff = 1.0 * tabEff[i] / tabEff[0];
        gra.fillRect(la * i, (int) (h - h * eff), la, h);//dessin de la ieme barre
        gra.drawString("" + (int) (eff * 100) + "%", la * i, (int) (h - h * eff));
    }// end for
}

```

Etape 4. Ajouter un bouton

Etape 5. Lier le bouton au PanelHisto

ActionListener=>method call (dessineHisto)

Etape 6. Lancement. Note.

Etape 7. Test. On affiche toujours le contenu du tableau par défaut. Changer les valeurs du tableau et observer les résultats et les bugs eventuels.