

DIGITAL BRIDGE

Leçon 10

But : Cette leçon fait usage des **fonctions** créées par le programmeur afin de faciliter la modification du code si le besoin se fait sentir plus tard. Le code Processing ci-dessous contient les fonctions suivantes :

- void setup()
- void draw()
- void afficheObjet(),
- void bougeObjet(),
- void bordure();

Nous connaissons déjà les fonctions habituelles void setup() et void draw(). Pour rappel void setup() sert à l'initialisation des variables et void draw() permet au programme de faire une loop infinie. Dans ce programme nous allons créer trois fonctions : void afficheObjet(), void bougeObjet(), bordure().

La taille de l'écran est définie dans la fonction void setup() et les trois autres fonctions sont incrustées dans la fonction void draw(){

```
background (255);  
afficheObjet() ;  
bougeObjet();  
bordures();  
}
```

Chaque fonction représente un bloc d'instruction qui peuvent être modifiées indépendamment. Les blocs d'instructions contenus dans les accolades { } sont exécutés dès que la fonction est invoquée (dans ce cas à l'intérieur de la fonction void draw()).

N.B. : Les fonctions sont définies en dehors de la loop void draw().

Comme exercice nous allons combiner dans la leçon 11 ce code Processing avec Arduino pour émettre un son chaque fois que l'objet touche une bordure.

Code Processing

```
float x = 0;  
float y = 0;  
float xspeed = 5;  
float yspeed = 2.3;  
void setup () {  
  size (640,640);  
}  
void draw(){  
  background (255);  
  afficheObjet() ;  
  bougeObjet();  
  bordures();  
}
```

```

void afficheObjet(){
    stroke(0);    // bordure de l'objet couleur noire
    fill(127);    // couleur de l'objet 127 = gris
    ellipse(x,y,32,32);    // forme de l'objet essayer aussi rect(x,y,32,32) ;
}

void bougeObjetl(){
    x = x +xspeed; // incremente la position x de l'objet
    y = y +yspeed; // incremente la position y de l'objet
}

void bordures() {
    if (x > width || x < 0) { // si x > que la largeur de l'ecran (x > 640) ou x < 0 alors change de direction
        //horizontal
        xspeed = xspeed * -1;
    }
    if (y > height || y < 0) { // si y > que la largeur de l'ecran (y > 640) ou y < 0 alors change de direction
        // vertical
        yspeed = yspeed * -1;
    }
}

```