# Ressources nécessaires :

* **OS :** Windows
* **Logiciel :** Processing 2
* **Editeur :** Notepad++
* **clientFTP :** Filezilla

**Objectif :** Découvrir et créer un jeu très simple, Pong, avec **processing** .

# Présentation de processing

Processing est une bibliothèque java et un environnement de développement libre (sous licence GNU GPL), créé par Benjamin Fry et Casey Reas, deux artistes américains. Processing est le prolongement « multimédia » de Design by numbers, l'environnement de programmation graphique développé par John Maeda au Media Lab du Massachusetts Institute of Technology.

Processing est tout particulièrement adapté à la création plastique et graphique interactive. Le logiciel fonctionne sur Macintosh, Windows, Linux, BSD et Android. Il est basé sur la plate-forme Java — il permet d'ailleurs de programmer directement en langage Java.



## Le cahier des charges

Pour créer notre Pong, on va tout d'abord dresser un cahier des charges :

* Le jeu sera seulement pour 1 joueur.
* Le terrain sera un carré noir de côté 400px coupé par une ligne blanche verticale en son milieu.
* Le plateau (là où rebondit la balle) sera un rectangle blanc de dimension 15px×85px.
* Il bougera avec les flèches UP et Down.
* La balle sera un cercle blanc de rayon 10px.
* La balle rebondira sur le plateau, sur le côté droit, sur le haut et le bas de la fenêtre.
* Si la balle sort du côté gauche, GAMEOVER s 'affiche.
* Pour rejouer une partie on appuyera sur la touche 'r'.

## La programmation

Aller dans **File → Save As...** et appeler votre projet pong.

# Les variables

Premièrement, on va initialiser les variables :

* x et y qui sont les coordonnées de la balle
* deplacementX et deplacementY qui s eront la direction de la balle
* w et z les coordonées du plateau

Le code :

int x, y;

int deplacementX, deplacementY;

int deplacementW, deplacementZ;

int w, z;

# Les fonctions prédéfinies

Ensuite, on crée notre métode void **setup**().

Nous allons donner des valeurs à ces variables :

* z = 60;
* w = 15;
* deplacementY = -3;
* deplacementX = 6;
* y = 200;
* x = 200;

On définit aussi la taille (400px×400px) de l'écran et la couleur noire du fond.

void **setup**()

{

size(400,400);

background(0);

x = 200;

y = 200;

deplacementX = int(random(4,8));

deplacementY = int(random(-4,-1));

w = 15;

z = 60;

}

Puis, on écrit le code de la méthode void **draw**() dans lequel nous allons appeler quatre sous-fonctions :

* nettoyer();
* dessiner();
* bouger();
* rebondir();

void **draw**()

{

nettoyer();

dessiner();

bouger();

rebondir();

}

# Les sous-fonctions

Ces quatre sous-fonctions n'ont pas de variables, on va donc utiliser l'instruction void.

void **nettoyer**()

{

// à compléter

}

void **dessiner**()

{

// à compléter

}

void **bouger**()

{

// à compléter

}

void **rebondir**()

{

// à compléter

}

La sous-fonction void **nettoyer**() permettra le rafraichissement de l'écran.

void **nettoyer**()

{

background(0);

}

La sous-fonction void **dessiner**() affichera les objets à l'écran.

void **dessiner**()

{

stroke(255);

fill(255);

rect(w,z,15,85);

fill(255);

ellipse(x,y,20,20);

stroke(255);

line(width/2,0,width/2,height);

}

La sous-fonction void **bouger**() animera la balle et le plateau.

void **bouger**()

{

x = x + deplacementX;

y = y + deplacementY;

if(keyPressed)

{

if(keyCode == DOWN && z+85<400)

{

z+=4;

}

if(keyCode == UP && z>0)

{

z-=4;

}

}

}

La sous-fonction void **rebondir**() gèrera les rebonds de la balle avec le plateau et les murs, ainsi que la fin de partie.

void **rebondir**()

{

// si on est trop à droite ET le déplacement horizontal est positif

if (x > width-10 && deplacementX > 0)

{

deplacementX = -deplacementX; // inverser la valeur

}

// si on est trop bas et le déplacement vertical est positif

if (y > width-10 && deplacementY > 0)

{

deplacementY = -deplacementY; // inverser la valeur

}

// si on est trop haut et le déplacement vertical est negatif

if (y < 10 && deplacementY < 10)

{

deplacementY = -deplacementY; // inverser la valeur

}

// si la balle rebondit sur le plateau

if (x < w+25 && x > w+10 && y+10 > z && y-10 < z+85)

{

// rendre positive cette valeur

deplacementX = abs(deplacementX);

}

// gestion de la fin de partie

if (x < -12)

{

deplacementX = 0;

deplacementY = 0;

fill(255,0,0);

textSize(28);

text("GAMEOVER",130,150);

textSize(18);

text("appuyer sur ' r ' pour rejouer",100,180);

if(keyPressed)

{

if(key == 'r')

{

x = 200;

y = 200;

deplacementX = int(random(4,8));

deplacementY = int(random(-4,-1));

}

}

}

}