(*) Comment créer un petit jeu vidéo avec Processing ?

> (*) : Il manque des informations. Cette page est en cours de construction et n'est donc pas finalisée.

Commençons par **choisir la taille de l'écran** ! 600 x 600 paraît pas mal pour commencer.

```
void setup() {
size( 600, 600 );
}
```

Bien, nous allons maintenant **créer notre joueur**. Ce sera... un gros pixel ! Nous allons lui donner **une position de départ, et référencer sa position**. Pour ce faire, nous utiliserons une variable de type *PVector()*, adaptée aux positions 2D. Puis nous allons **le dessiner**

```
PVector position_initiale;
PVector position_actuelle;

void setup() {
  size( 600, 600 );

  position_initiale = new PVector( 20, 20 );
  position_actuelle = position_initiale;
}

void draw() {
  rect( position_actuelle.x - 10,
  position_actuelle.y - 10,
  20,
  20 );
}
```

Parfait ! Maintenant, nous allons **le faire réagir au clavier** :

```
PVector position_initiale;
PVector position_actuelle;

int vitesse_deplacement = 4;

void setup() {
  size( 600, 600 );

  position_initiale = new PVector( 20, 20 );
  position_actuelle = position_initiale;
}

void draw() {
  rect( position_actuelle.x - 10,
  position_actuelle.y - 10,
  20,
  20 );
}



void keyPressed() {
  if (key == CODED) {
      if (keyCode == UP) {
        position_actuelle.y = position_actuelle.y - vitesse_deplacement;
      };
      if (keyCode == DOWN) {
        position_actuelle.y = position_actuelle.y + vitesse_deplacement;
      };

      if (keyCode == LEFT) {
        position_actuelle.x = position_actuelle.x - vitesse_deplacement;
      };
      if (keyCode == RIGHT) {
        position_actuelle.x = position_actuelle.x + vitesse_deplacement;
      };

  };
}
```

Cela fonctionne... mis-à-part **trois problèmes** !

Premièrement, **les anciennes positions du personnages sont encore affichées à l'écran**. Cela se règle simplement, en repeignant l'écran entre chaque déplacement, en changeant la fonction *draw()* comme ceci :

```
void draw() {
  background( 255 );
  rect( position_actuelle.x - 10,
  position_actuelle.y - 10,
  20,
  20 );
}
```

Ensuite, le personnage se déplace bizarrement. C'est parce que l'algorithme lié au clavier choisi est le mauvais. C'est dans la fonction draw() que l'on peut régler cela :

```
void draw() {
  background( 255 );

  if (keyPressed == true) {
    if (key == CODED) {
      if (keyCode == UP) {
        position_actuelle.y = position_actuelle.y - vitesse_deplacement;
      };
      if (keyCode == DOWN) {
        position_actuelle.y = position_actuelle.y + vitesse_deplacement;
      };

      if (keyCode == LEFT) {
        position_actuelle.x = position_actuelle.x - vitesse_deplacement;
      };
      if (keyCode == RIGHT) {
        position_actuelle.x = position_actuelle.x + vitesse_deplacement;
      };

    };
  };

  rect( position_actuelle.x - 10,
  position_actuelle.y - 10,
  20,
  20 );
```

```
}
```

Enfin, il est possible de sortir de l'écran... Voici la solution :

```
void draw() {
  background( 255 );
  rect( position_actuelle.x - 10,
  position_actuelle.y - 10,
  20,
  20 );
}
```

Ce qui donne à ce stade le code suivant :

```
PVector position_initiale;
PVector position_actuelle;

int vitesse_deplacement = 4;

void setup() {
  size( 600, 600 );

  position_initiale = new PVector( 20, 20 );
  position_actuelle = position_initiale;
}

void draw() {
  background( 255 );

  if (keyPressed == true) {
    if (key == CODED) {
      if (keyCode == UP) {
        position_actuelle.y = position_actuelle.y - vitesse_deplacement;
      };
      if (keyCode == DOWN) {
        position_actuelle.y = position_actuelle.y + vitesse_deplacement;
      };

      if (keyCode == LEFT) {
        position_actuelle.x = position_actuelle.x - vitesse_deplacement;
```

```
      };
      if (keyCode == RIGHT) {
        position_actuelle.x = position_actuelle.x + vitesse_deplacement;
      };

    };
  };

  if ( position_actuelle.x < 0 ) { position_actuelle.x = position_actuelle.x + 600; };
  if ( position_actuelle.x > 600 ) { position_actuelle.x = position_actuelle.x - 600; };
  if ( position_actuelle.y < 0 ) { position_actuelle.y = position_actuelle.y + 600; };
  if ( position_actuelle.y > 600 ) { position_actuelle.y = position_actuelle.y - 600; };

  rect( position_actuelle.x - 10,
  position_actuelle.y - 10,
  20,
  20 );
}
```

[EN CONSTRUCTION]

Révision #3
Créé 2023-08-02 15:18:17 CEST par Rachelle
Mis à jour 2024-12-17 14:01:07 CET par Rachelle