

## 6.1.∞ Le module p5

### 1. 1. C'est quoi, le module p5 ?

En Python, un module est un fichier constitué de code Python, qui contient en général des fonctions dédiées à traiter un thème bien précis. Il en existe énormément, ce qui participe à la popularité de ce langage.

Par exemple, vous avez certainement utilisé en seconde<sup>1</sup> les modules `math` (qui contient en autres les fonctions mathématiques usuelles) et `random` (qui contient des fonctions de génération de nombres pseudo-aléatoires). Nous utiliserons bien d'autres modules au cours de l'année, pour traiter des images, des données csv, faire des requêtes web, etc.

Nous étudierons ces modules dans un chapitre à part un peu plus tard dans l'année.

Le module p5 de Python est un portage du langage **Processing**, qui est un outil de création multimédia utilisant le code informatique. Simple de prise en main, il a été créé par des artistes pour des artistes.

Nous l'utiliserons pour ajouter du graphisme à nos créations...

## Squelette de programme avec p5 (obligatoire!)

### Script Python

```

1  from p5 import *
2
3  def setup():
4      # corps de la fonction à compléter
5
6
7  def draw():
8      # corps de la fonction à compléter
9
10 run()
```

#### Ligne 1

On importe le module, c'est-à-dire qu'on charge en mémoire tout le contenu du fichier.

#### la fonction `setup`

Elle contient le paramétrage initial du programme (taille de la fenêtre, couleur de fond, etc).

#### la fonction `draw`

Le contenu de cette fonction est continuellement exécuté, à raison de 60 images par seconde par défaut. Sauf si la fonction `no_loop` est appelée dans `setup`.

#### la fonction `run`

Elle lance la fenêtre d'exécution du programme.

## 2. 2. Les bases

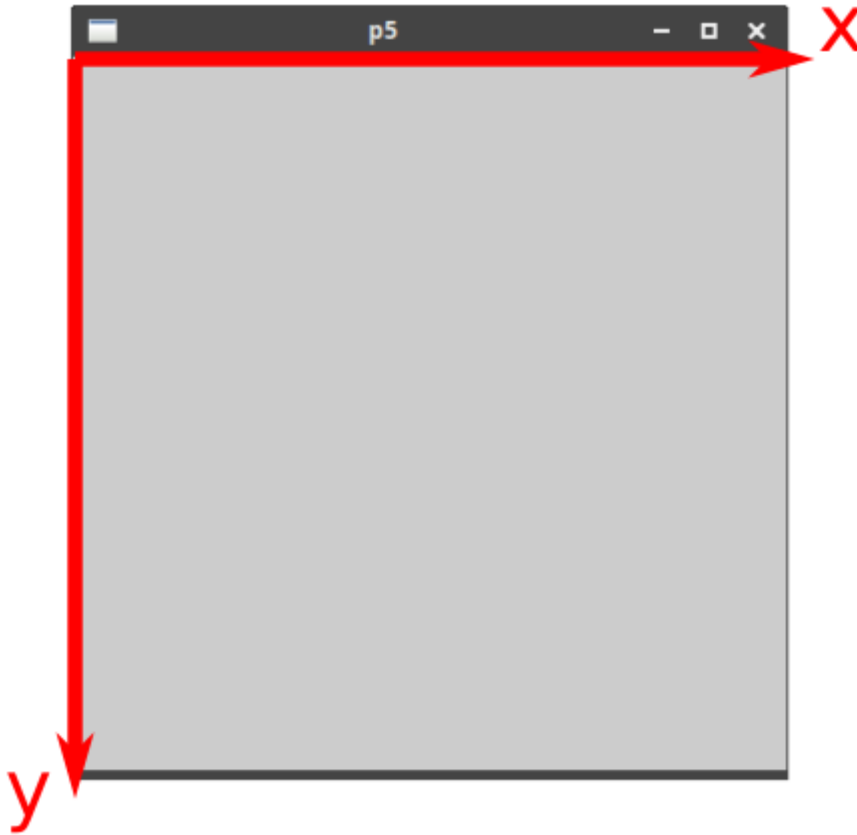
### 2.1. 2.1 Repère

À l'exécution de la fonction `run`, une fenêtre s'affiche avec une zone de dessin qu'on appelle généralement *canevas* (canvas in english). Sa taille se définit à l'aide de la fonction `size` à l'intérieur de la fonction `setup`. Par exemple, pour créer une zone de dessin de 300 pixels sur 200 pixels, on utilisera:

### Script Python

```
size(300, 200)
```

Chaque pixel de cette zone est repérée par des coordonnées dans le repère suivant, dont l'origine se situe en haut à gauche et l'axe des ordonnées est **orienté vers le bas**.



### 2.2. 2.2 Tracés

#### i Tracés de base

- `point` : permet de dessiner un point (pixel). En paramètre, les coordonnées du point.
- `line` : permet de tracer une ligne entre deux points. En paramètres, les coordonnées des deux points.
- `rect` : permet de tracer un rectangle. En paramètres, les coordonnées du sommet haut-gauche, puis la largeur et la hauteur du rectangle.
- `ellipse` : permet de tracer une ellipse. En paramètres, les coordonnées du centre, puis la largeur et la hauteur (mettre la même valeur pour un cercle).

## Exemples

Copier-coller le code suivant et faire varier les paramètres pour bien les comprendre.

### Script Python

```
from p5 import *

def setup():
    size(200, 200)
    no_loop()

def draw():
    rect((90, 50), 20, 100)
    ellipse((100, 70), 60, 60)
    ellipse((81, 70), 16, 32)
    point(81, 70)
    ellipse((119, 70), 16, 32)
    point(119, 70)
    line((90, 150), (80, 160))
    line((110, 150), (120, 160))
run()
```

## 2.3.2.3 Couleurs

### Pinceau

- `background` : permet de définir la couleur du fond de la zone de dessin. En paramètres, les trois composantes RGB de la couleur.
- `stroke` : permet de définir la couleur du pinceau (noir par défaut) pour le contour de la forme. En paramètres, les trois composantes RGB de la couleur.
- `no_stroke()` : permet de dessiner une forme sans contour (pas de paramètre).
- `stroke_weight` : permet de définir la largeur du pinceau. En paramètre, le nombre de pixels.
- `fill` : permet de définir la couleur de remplissage de la forme. En paramètres, les trois composantes RGB de la couleur.

## Exemples



Copier-coller le code suivant et faire varier les paramètres pour bien les comprendre.

### Script Python

```
from p5 import *

def setup():
    size(200, 200)
    no_loop()

def draw():
    background(12, 12, 12)
    fill(255, 0, 0)
    rect((90, 50), 20, 100)
    no_stroke()
    fill(244, 164, 96)
    ellipse((100, 70), 60, 60)
    stroke(1)
    fill(255, 255, 255)
    ellipse((81, 70), 16, 32)
    point(81, 70)
    ellipse((119, 70), 16, 32)
    stroke_weight(5)
    point(119, 70)
    stroke_weight(2)
    line((90, 150), (80, 160))
    line((110, 150), (120, 160))
run()
```

## 3. 3. Exercice

### Exercice 1

Dans un canevas de 300 x 200 pixels, dessiner successivement les drapeaux français, japonais puis tchèque.

Consulter la documentation pour des triangles ou polygones.

1. bon ok, ça dépend de quel prof vous avez eu l'an dernier... ←