

# Mini-Projet : Dessine ma rue (avec Turtle)

## Présentation du Projet

### But

On souhaite écrire un programme qui permet de générer « aléatoirement » le dessin d'une rue de 4 immeubles. On utilisera pour cela le module Turtle de Python.



### Contraintes

Les contraintes urbanistiques sont les suivantes :

- Les immeubles ont au minimum un rez-de-chaussée et au maximum 4 étages (5 niveaux) ;
- Les immeubles ont une largeur de 140 pixels ;
- Les immeubles ont une couleur unique pour toute la façade ;
- Chaque niveau (rez-de-chaussée ou étage) a une façade de hauteur 60 pixels ;
- Les rez-de-chaussée n'ont qu'une seule porte et 2 fenêtres placées aléatoirement ;
- Toutes les fenêtres sont identiques, de taille 30 pixels sur 30 pixels ;
- Toutes les portes fenêtrées ont un balcon et font une taille de 30 pixels en largeur et 50 pixels en hauteur ;
- Le toit peut avoir 2 formes : plat ou triangulaire ;
  - Si le toit est plat : il fait une épaisseur de 10 pixels ;
  - Si le toit est triangulaire, il fait une hauteur de 40 pixels pour une base de 160 pixels.

## Exemples

La série d'exemples ci-dessous est basée sur :

- une couleur aléatoire pour les façades ;
- deux modèles de toits ;
- deux modèles de portes avec une couleur aléatoire ;
- deux modèles d'ouvertures pour les étages : fenêtre ou porte-fenêtre avec balcon ;
- trois éléments horizontalement pour chaque niveau.



# Travail à faire

Proposer un programme sous forme de modules qui réponde au problème posé en utilisant **le module Turtle de Python (voir memento joint)**.

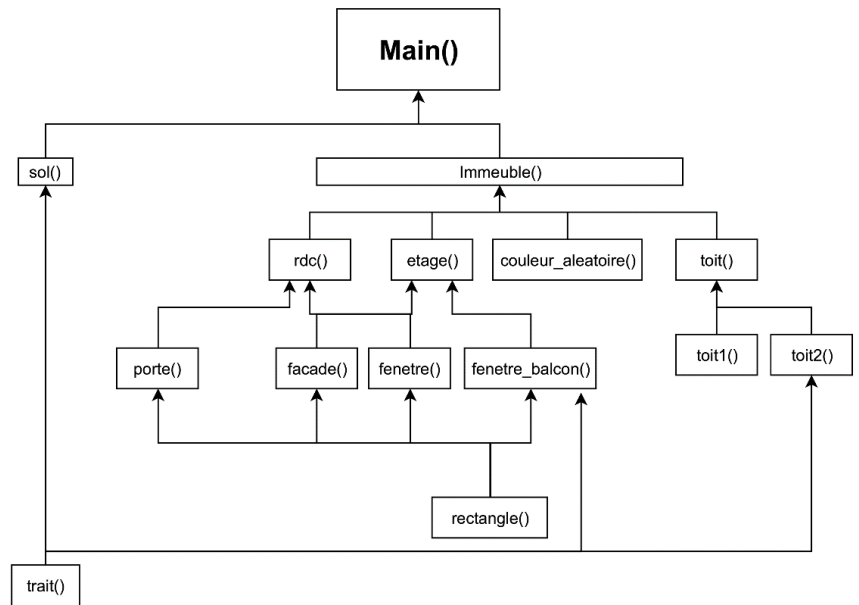
Vous utiliserez donc le plus de petites fonctions possible comme décrit ci-contre.

Vous travaillerez collectivement et en interdépendance à travers des importations de modules.

Pour cela on a recensé toutes les fonctions à écrire. Leurs dépendances sont décrites dans le schéma ci-contre.

Vous devez donc vous répartir les modules à écrire en commençant par ceux du bas car il ne dépende pas des autres.

Pour vous aider dans votre tâche, dans chaque module, les importations et les doc-strings ont déjà été faits ainsi que les commentaires. Il ne vous reste qu'à écrire le code en ayant bien compris ce que votre module importe avant de commencer.



## Fonction immeuble(x, y\_sol)

x : abscisse du centre de l'étage

y\_sol : ordonnée du sol du la rue

## Fonction sol(y\_sol):

y\_sol : ordonnée du sol du la rue

## Fonction rdc(x, y\_sol, c\_facade, c\_porte) :

x : (int) abscisse du centre

y\_sol : ordonnée du sol du la rue

c\_facade : couleur de la façade

c\_porte : couleur de la porte

## Fonction etage(x, y\_sol, couleur, niveau) :

x : abscisse du centre de l'étage

y\_sol : ordonnée du sol du la rue

couleur : couleur de la façade de l'étage

niveau : numéro de l'étage en partant de 0 ( RDC)

## Fonction couleur\_aleatoire():

Cette fonction retourne un triplet de 3 nombres entiers compris entre 0 et 255 correspondant à du RGB

## Fonction toit(x, y\_sol, niveau):

x : abscisse du centre de l'étage

y\_sol: ordonnée du sol

niveau : numéro de l'étage en partant de 0 (RDC)

## Fonction porte(x,y,couleur):

x est l'abscisse du centre de la porte

y est l'ordonnée du sol du niveau de la porte

## Fonction facade(x, y\_sol, couleur, niveau):

x : abscisse du centre de la façade

y\_sol : ordonnée du sol du la rue

couleur : couleur de la façade

niveau : num du niveau (0 pour les rdc, ...)

## Fonction fenetre(x,y):

x est l'abscisse du centre de la fenêtre

y est l'ordonnée du sol du niveau de la fenêtre

## Fonction fenetre\_balcon(x,y):

x est l'abscisse du centre de la porte-fenêtre-balcon

y est l'ordonnée du sol du niveau de la porte-fenetre-balcon

## Fonction toit1(x, y\_sol, niveau):

x : abscisse du centre du toit

y\_sol : ordonnée du sol du la rue

Niveau : num du niveau (0 pour les rdc, ...)

## Fonction toit2(x, y\_sol, niveau):

x : abscisse du centre du toit

y\_sol : ordonnée du sol du la rue

niveau : num du niveau (0 pour les rdc, ...)

## Fonction rectangle(x,y,w,h):

x, y : coordonnées du centre de la base de rectangle

w : largeur du rectangle

h : hauteur du rectangle

## Fonction trait(x1,y1,x2,y2):

x1, y1 : coordonnées du début du trait

x2, y2 : coordonnées de la fin du trait

**Remarques :** Dans le dossier, vous trouverez des images de ce que chaque module peut faire individuellement (et à partir des modules importés). Attention aussi à ne pas réinventer tout à chaque module mais à bien **utiliser les fonctionnalités des modules importés par le module que vous écrivez**.