

TP Modularité

Objectifs :

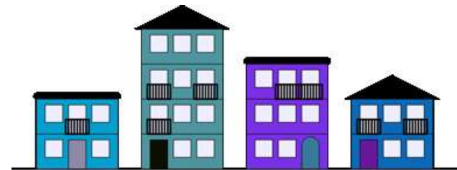
- ▷ Utiliser des bibliothèques
- ▷ Exploiter leur documentation
- ▷ Créer des modules simples et les documenter

1 Présentation du TP

1.1 But

On souhaite écrire un programme qui permet de générer aléatoirement le dessin d'une rue de 4 immeubles.

On utilisera pour cela la bibliothèque *Turtle* de Python.



1.2 Contraintes

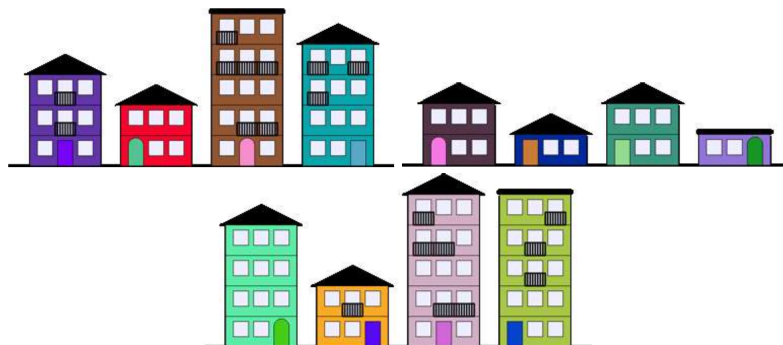
Les contraintes urbanistiques sont les suivantes :

- ▷ 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 fenêtre ont un balcon et font une taille de 30 pixels en largeur et 50 pixels en hauteurs
- ▷ Le toit peut avoir 2 formes :
 - soit plat : il fait une épaisseur de 10 pixels
 - soit triangulaire : il fait une hauteur de 40 pixels pour une base de 160 pixels

1.3 Exemples

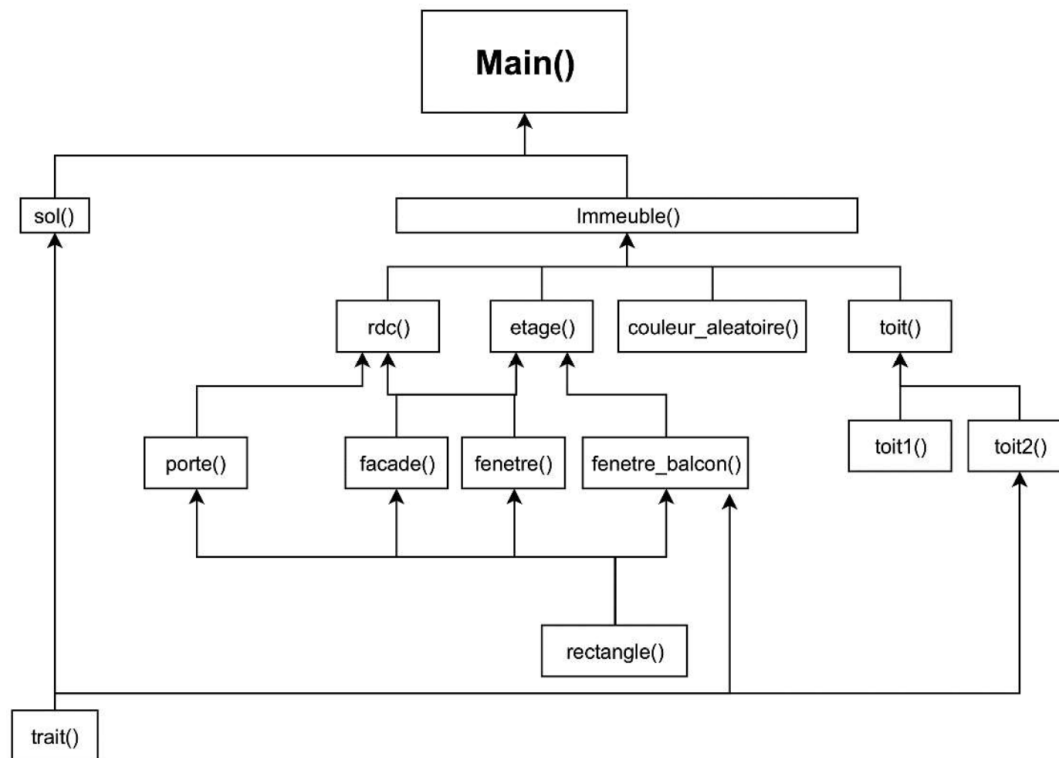
La série d'exemples ci-dessous est basée sur :

- ▷ 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.



2 Travail à faire

- ▷ Proposer un programme sous forme de modules qui réponde au problème posé en utilisant la bibliothèque Turtle de Python (voir memento joint).
- ▷ 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.



- ▷ Utiliser le plus de petites fonctions possibles comme décrit dans le schéma ci-dessus.
- ▷ Se répartir les modules à écrire en commençant par ceux du bas car ils ne dépendent pas des autres.
- ▷ Pour vous aider dans votre tâche, dans chaque module, les **importations** et les **docstrings** ont déjà été faits ainsi que les **commentaires**.

Exemple :

```

1 def porte(x,y,couleur):
2     ''' Parametres :
3         x est l'abscisse du centre de la porte
4         y est l'ordonnee du sol du niveau de la porte
5         couleur : couleur de la porte
6     Remarque:
7         Cette fonction dessine une porte de 30 pixels de large
8         pour 50 pixels de hauteur
9     '''
10    pass
  
```

- ▷ Ecrire le code manquant pour chacun des modules afin de générer la rue demandée en ayant bien compris ce que votre module importe avant de commencer.

Remarque :

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**.