Rappel : méthodes `setup` et `draw`

Le code ci-dessous permet de tracer des lignes sans arrêt (point de départ : le centre de la toile)

```
1 void setup(){
2     println("Salut toto");
3     size(600,600);
4 }
5 void draw(){
6     line(300,300,random(600),random(600));
7     println("rebonjour toto");
8 }
```

On peut dire qu'une méthode permet de donner un nom à plusieurs lignes de code

1. les lignes 2-3 : un bloc de code appelé `setup()`
2. les lignes 6-7 : un bloc de code appelé `draw()`

Rappel : méthodes `setup` et `draw`

Le code ci-dessous permet de tracer des lignes sans arrêt (point de départ : le centre de la toile)

```
1 void setup(){
2     println("Salut toto");
3     size(600,600);
4 }
5 void draw(){
6     line(300,300,random(600),random(600));
7     println("rebonjour toto");
8 }
```

On peut dire qu'une méthode permet de donner un nom à plusieurs lignes de code

1. les lignes 2-3 : un bloc de code appelé `setup()`
2. les lignes 6-7 : un bloc de code appelé `draw()`

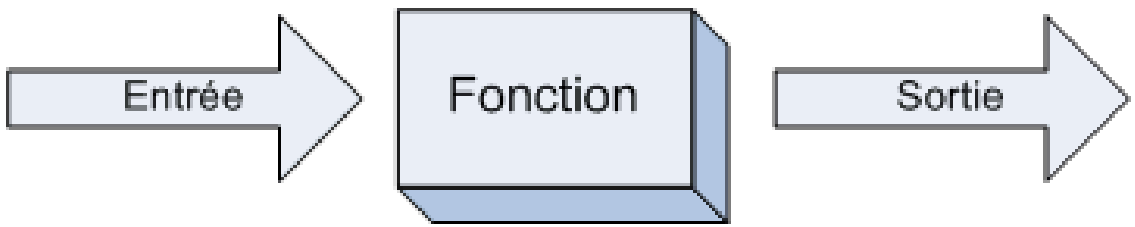
Fonctions

Une fonction comporte :

une entrée on fait « rentrer » des informations dans la fonction (on lui donne les données avec lesquelles travailler).

un traitement grâce aux informations reçues en entrée, la fonction exécute un traitement, elle fait quelque chose

une sortie à la fin la fonction renvoie un résultat via return.



méthode = une fonction qui ne renvoie rien (void)

- Une méthode n'a pas besoin de return
- void= pas de sortie

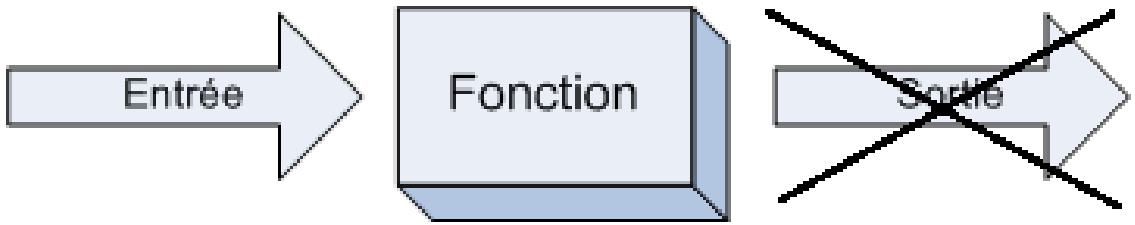
Fonctions

Une fonction comporte :

une entrée on fait « rentrer » des informations dans la fonction (on lui donne les données avec lesquelles travailler).

un traitement grâce aux informations reçues en entrée, la fonction exécute un traitement, elle fait quelque chose

une sortie à la fin la fonction renvoie un résultat via return.



méthode = une fonction qui ne renvoie rien (void)

- Une méthode n'a pas besoin de return
- void= pas de sortie