

L'instruction

```
1 if __name__ == "__main__":
```

ne s'exécutera que si le fichier est lancé *directement*. S'il est appelé comme un module dans un autre fichier, ces instructions sont omises. C'est une méthode pour développer des tests quand on crée un module.

### Activité 1 :

```
1 class Noeud:
2     def __init__(self,e,s):
3         self.donnees = e
4         self.successeur = s
5
6
7 class Pile:
8     def __init__(self):
9         self.dernier = None
10
11     def est_vide(self):
12         return self.dernier is None
13
14     def empiler(self,e):
15         self.dernier = Noeud(e,self.dernier)
16
17     def depiler(self):
18         if not self.est_vide():
19             res = self.dernier.donnees
20             self.dernier = self.dernier.successeur
21             return res
22
23     def __str__(self):
24         affiche = ""
25         last = self.dernier
26         while last is not None:
27             affiche += str(last.donnees) + "\n"
28             last = last.successeur
29         return affiche
30
31 if __name__ == "__main__":
32     p = Pile()
33     p.empiler(3)
34     p.empiler(9)
35     p.empiler(4)
36     print(p)
37     print("dépile:",p.depiler())
38     print(p)
```

### Activité 2 :

```
1 class Noeud():
2     def __init__(self,e,s):
3         self.donnees = e
```

```
4         self.successeur = s
5
6     def __str__(self):
7         return str(self.donnees) + '|'
8
9 class File():
10     def __init__(self):
11         self.premier = None
12         self.dernier = None
13
14     def est_vide(self):
15         return self.premier == None
16
17     def enfiler(self,e):
18         if self.est_vide():
19             # 1 seul élément: le premier est le dernier également
20             self.premier = Noeud(e,None)
21             self.dernier = self.premier
22         else:
23             nouveau = Noeud(e,None)
24             # le dernier devient avant-dernier
25             self.dernier.successeur = nouveau
26             # le nouveau devient dernier
27             self.dernier = nouveau
28
29     def defiler(self):
30         if not self.est_vide():
31             res = self.premier.donnees
32             self.premier = self.premier.successeur
33             return res
34
35     def __str__(self):
36         c = self.premier
37         s = ''
38         while not c is None:
39             s = s + c.__str__()
40             c = c.successeur
41         return '\u2BA4|' + s[:] + '\u2BA0'
42
43 if __name__ == "__main__":
44     from random import randint
45
46     a = File()
47     for i in range(6):
48         a.enfiler(randint(1, 6))
49         print(a)
50
51     for i in range(6):
52         a.defiler()
53         print(a)
```