

# Boucle bornée (for)

La boucle "for" permet de répéter un nombre de fois connu des instructions

Langage naturel	Python
Pour $i$ allant de 3 à 7 Instructions Fin Pour	<code>for k in range(3,8):</code> Instructions

## Exercice:

Ecrire un programme qui calcul la somme  $S=1+2+3+....+n-1+n$  des  $n$  premiers entiers:

Langage naturel	Python
$S \leftarrow 0$ Pour $k$ variant de 1 à $n$ $S \leftarrow S+k$ Fin pour	<code>S=0</code> <code>for k in range (1,n+1):</code> <code>S=S+k</code>

## Exemple pour $n=4$ :

The screenshot shows a Python IDE window with the following code:

```
1 S=0
2 for k in range(1,5):
3     S=S+k
```

Annotations (callouts) point to specific parts of the code:

- A callout points to the range function: "Il faut augmenter la valeur de fin de 1" (The end value must be increased by 1).
- A callout points to the colon at the end of the for loop line: "Ne pas oublier les « : »" (Don't forget the « : »).
- A callout points to the indentation of the loop body: "Indenter absolument les instructions à l'intérieur de la boucle" (Absolutely indent the instructions inside the loop).

Below the code editor, the Python shell output is visible:

```
>>>
>>> S
10
```

	k	S	
<b>Avant le début de la boucle</b>		0	Au début, la variable S est initialisée à 0
<b>Fin de la 1<sup>ère</sup> Itération</b>	1	1	k=1, on ajoute 1 à S qui vaut maintenant 1
<b>Fin de la 2<sup>ème</sup> Itération</b>	2	1+2=3	k=2, on ajoute 2 à S qui vaut maintenant 1+2=3
<b>Fin de la 3<sup>ème</sup> Itération</b>	3	3+3=6	k=3, on ajoute 3 à S qui vaut maintenant 3+3=6
<b>Fin de la 4<sup>ème</sup> Itération</b>	4	6+4=10	k=4, on ajoute 4 à S qui vaut maintenant 6+4=10 On sort de la boucle