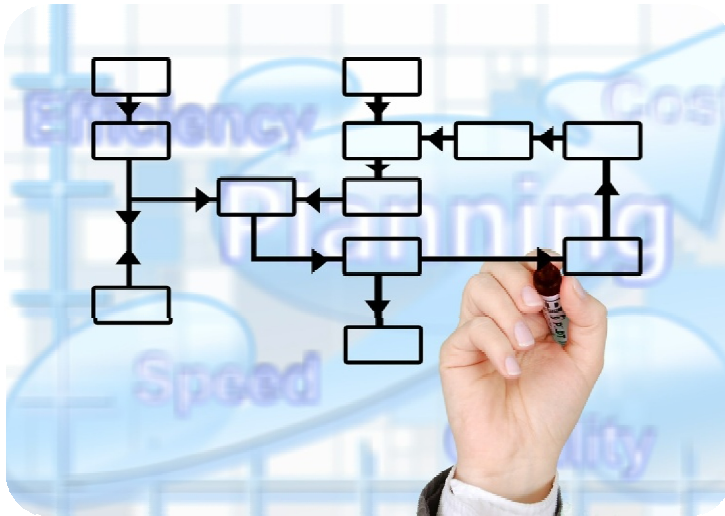


A stylized illustration of a computer monitor with a black bezel and a grey base. The screen is white and displays the title text.

Les structures répétitives

Les structures répétitives

Objectifs : savoir utiliser les structures permettant de répéter des actions.



- 1- Tant que et Faire...tant que
- 2- La structure répétitive POUR
- 3- Comparaison des boucles

Itérer

1 Les boucles classiques

Rappelons que l'enchaînement **répétitif** consiste à exécuter certaines actions aussi longtemps qu'une condition préalable est vérifiée, ou à exécuter certaines actions jusqu'à ce qu'une condition pré-établie soit vérifiée.

- Structure TANT QUE

TANT QUE <condition>



Exécute les actions suivantes

FIN-TANT- QUE

- Structure REPETER

REPETER



Les actions suivantes

JUSQU'A <condition>

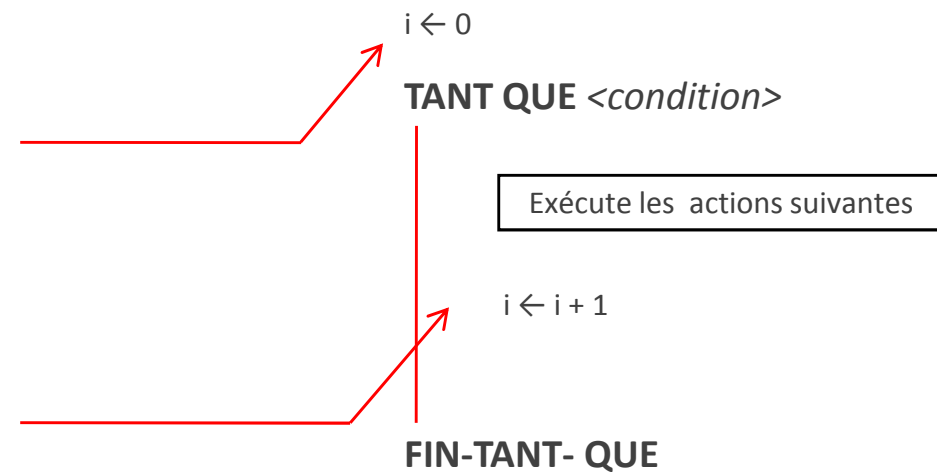
Itérer

2 Les boucles classiques : attention

Pour ces 2 boucles, vous devez prendre des précautions pour que tout se passe bien. En effet, il ne faut pas être enfermée dans la boucle à un moment donné. **Ces 2 boucles sont conseillées quand on ne sait pas a priori combien de fois itérer.**

● Structure TANT QUE

1. La **<condition>** doit être vraie pour entrer.
2. Pour rendre la condition vraie, **on initialise souvent une variable** dont dépend la condition juste avant la boucle.
3. Pour ne pas s'enfermer dans la boucle, la **<condition> doit être fausse** à un moment donné.
4. Pour faire bouger la **<condition>**, il est courant **d'incrémenter** ou **décrémenter** la variable initialisée.



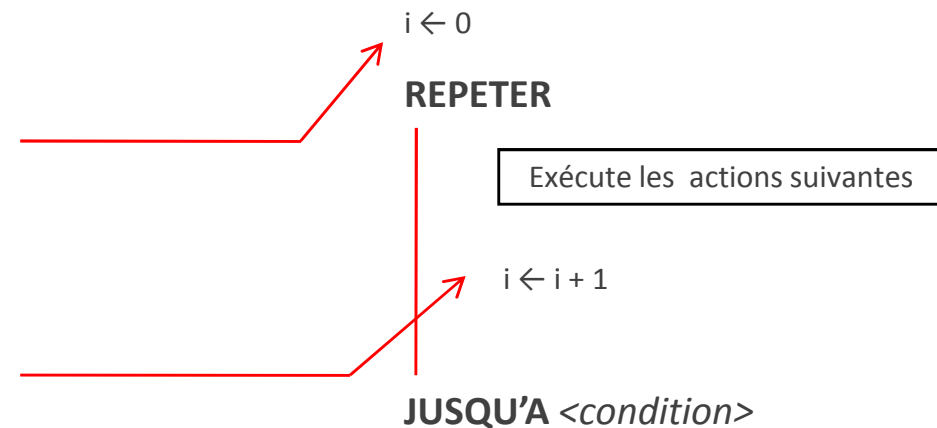
Itérer

2 Les boucles classiques : attention (suite)

Pour ces 2 boucles, vous devez prendre des précautions pour que tout se passe bien. En effet, il ne faut pas être enfermée dans la boucle à un moment donné. **Ces 2 boucles sont conseillées quand on ne sait pas a priori combien de fois itérer.**

● Structure REPETER

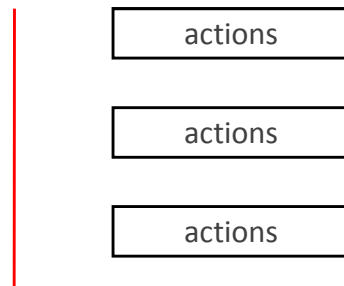
1. On entre dans la boucle sans vérifier une **<condition>** d'entrée.
2. Il est aussi possible **d'initialiser une variable** avant d'entrer dans la boucle.
3. Pour ne pas s'enfermer dans la boucle, la **<condition>** doit être vraie à un moment donné.
4. Pour faire bouger la **<condition>**, il est courant **d'incrémenter** ou **décrémenter** la variable initialisée.



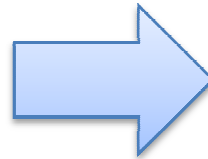
*Itérer***1** Itérer un certain nombre de fois

Si on sait combien de fois on va itérer, il est conseillé d'utiliser la boucle POUR. On itère selon un indice variant d'une valeur initiale à une valeur finale, en progressant de *inc* en *inc*, *inc* étant souvent égal à 1.

POUR indice **DE** init **A** fin **PAS DE** inc **FAIRE**



FIN-POUR



```
nb_pages ← 100
```

```
POUR no DE 1 A nb_pages
```

```
  Lire la page
```

```
FIN-POUR
```

ou bien

```
nb_pages ← 100
```

```
no ← 1
```

```
TANT QUE no <= nb_pages
```

```
  Lire la page
```

```
  no ← no + 1
```


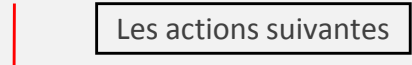
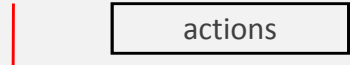
```
FIN-TANT-QUE
```

Attention :

En général, ni la variable **indice**, ni la **borne finale**, ne doivent être modifiés par la ou les instructions du bloc, sous peine de résultats imprévisibles.

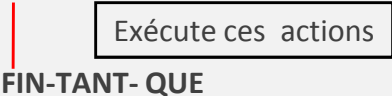
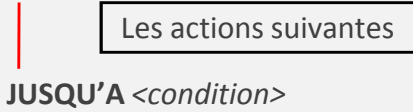
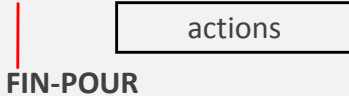
Itérer

1 Tableau comparatif

TANT QUE <i><condition></i>  FIN-TANT- QUE	REPETER  JUSQU'A <i><condition></i>	POUR indice DE init A fin PAS DE inc FAIRE  FIN-POUR
<ul style="list-style-type: none"> Exécution si et seulement si <condition> vraie. <condition> doit être fausse pour sortir de la boucle. On ne sait pas <i>a priori</i> combien de fois on va boucler. 	<ul style="list-style-type: none"> Exécution au moins une fois. <condition> doit être vraie pour sortir de la boucle. On ne sait pas <i>a priori</i> combien de fois on va boucler. 	<ul style="list-style-type: none"> L'indice varie de init à fin. indice est augmenté de inc à chaque tour de boucle. Le nombre d'itérations est connu d'avance.

Itérer

1 Tableau comparatif (suite)

TANT QUE <condition>  FIN-TANT-QUE	REPETER  JUSQU'À <condition>	POUR indice DE init A fin PAS DE inc FAIRE  FIN-POUR
Programme Exemple VAR n, cpt : Entier (* n : la boucle s'arrête à n cpt : c'est le compteur *) DEBUT ECRIRE 'Saisir n' LIRE n cpt ← 1 TANT QUE cpt ≤ n ECRIRE cpt cpt ← cpt + 1 FIN-TANT-QUE FIN	Programme Exemple VAR n, cpt : Entier (* n : la boucle s'arrête à n cpt : c'est le compteur *) DEBUT ECRIRE 'Saisir n' LIRE n cpt ← 0 REPETER cpt ← cpt + 1 ECRIRE cpt JUSQU'À cpt = n FIN	Programme Exemple VAR n, cpt : Entier (* n : la boucle s'arrête à n cpt : c'est le compteur *) DEBUT ECRIRE 'Saisir n' LIRE n POUR cpt DE 1 A n FAIRE ECRIRE cpt FIN-POUR FIN