XWIKI NATIS

Algorithme #1

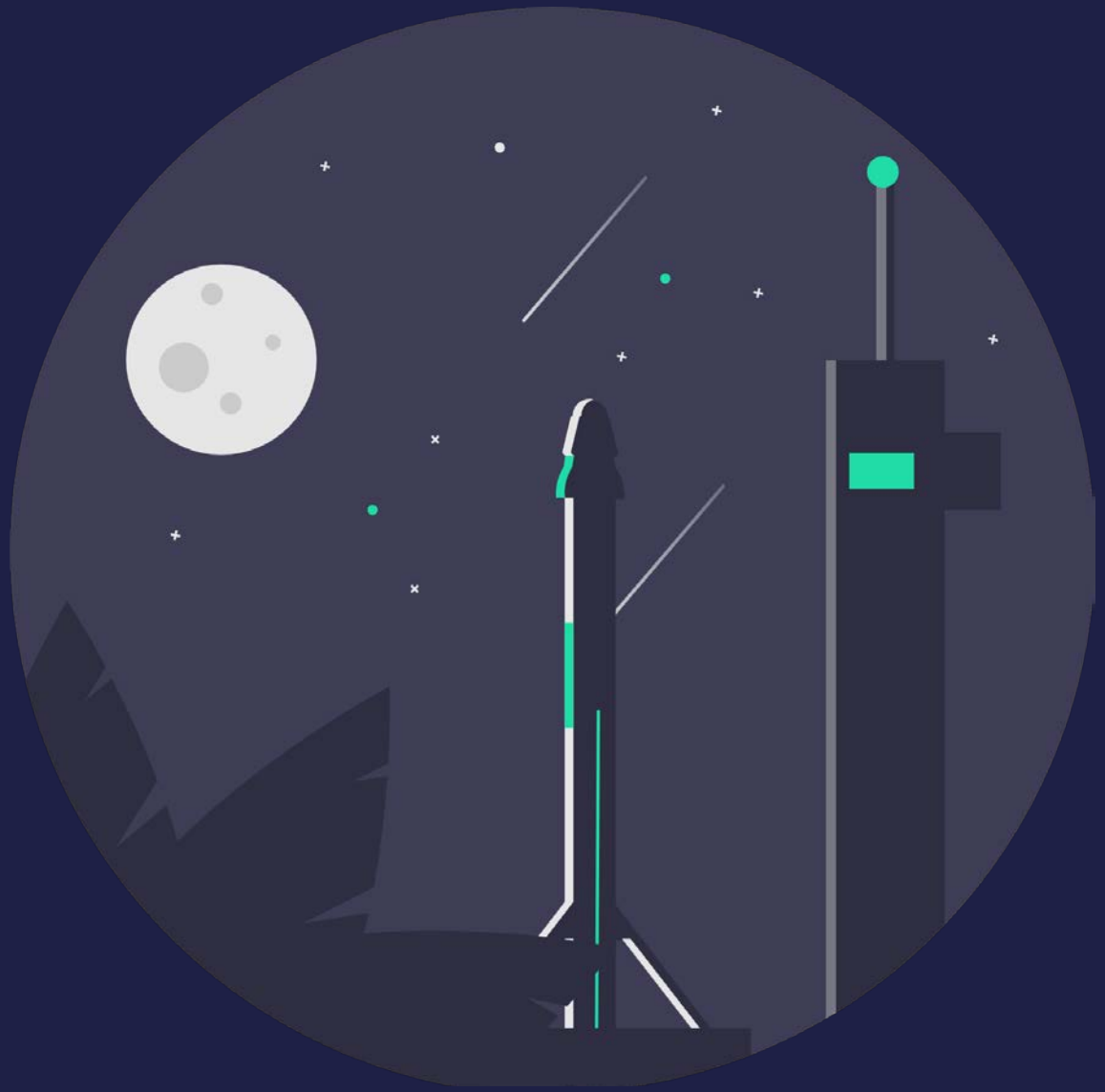
Michael

XWIKI NATIS

XWIKI NATIS

Blocs
Boucles
Conditions

XDNATIS



XDNATIS

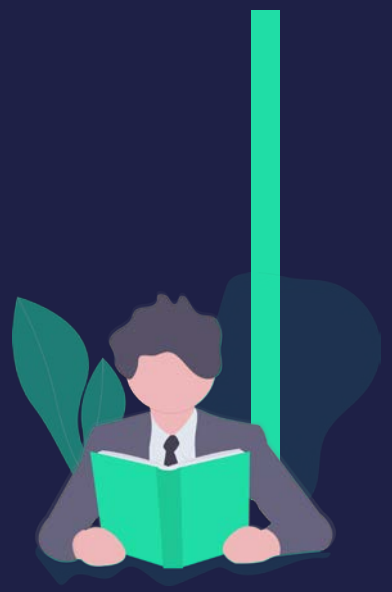
Compétence demandée :
Maîtriser les 5 concepts de la construction algorithmique

XWIKI NATIS

1. Variables
2. Instructions de base
3. Blocs
4. Conditions
5. Boucles

XWIKI NATIS

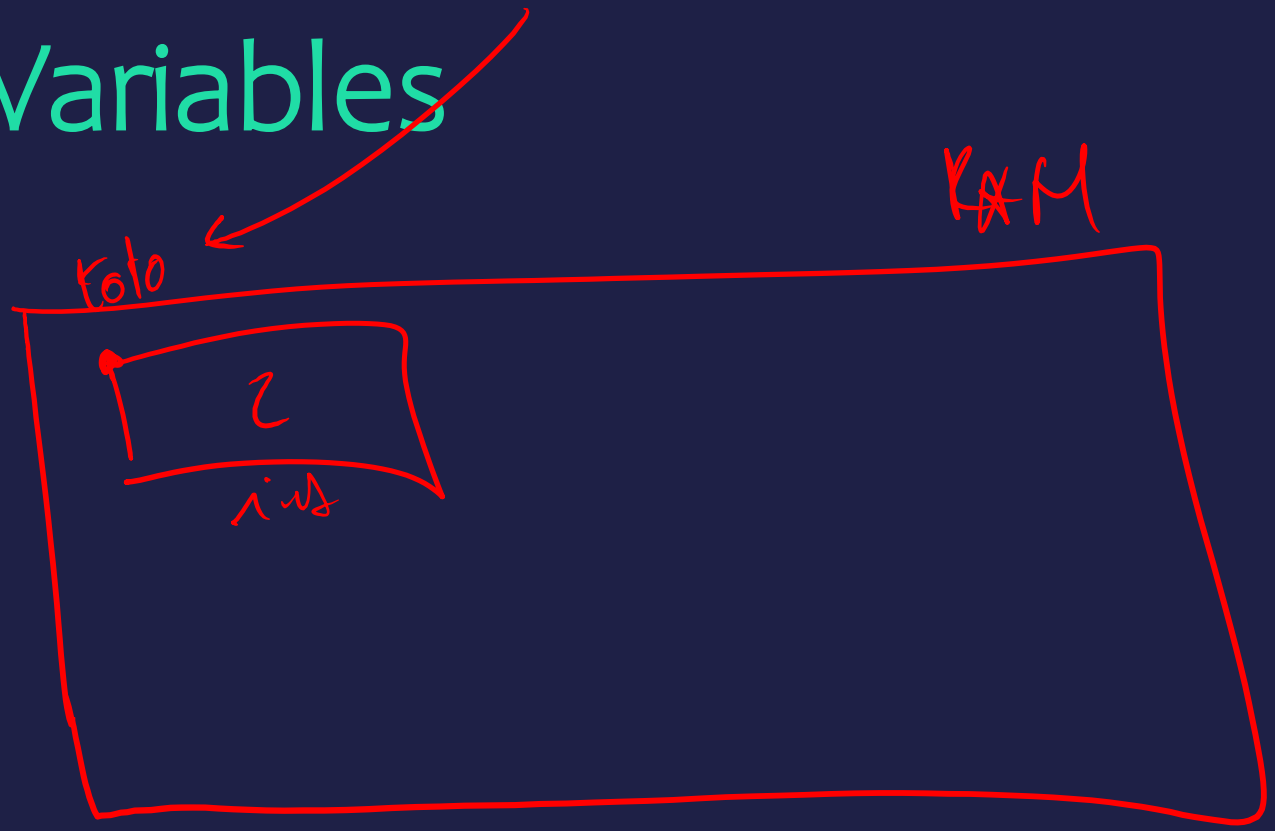
1. Variables
2. Instructions de base
3. Blocs
4. Conditions
5. Boucles



XWIKI SAS

XEBIANS

1. Variables



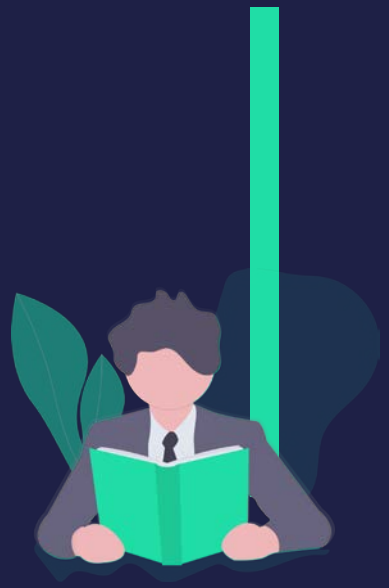
XWIKI NATIS

Les variables sont typées !

XWIKI NATIS

Les variables sont TYPÉES !

XWIKI NATIS

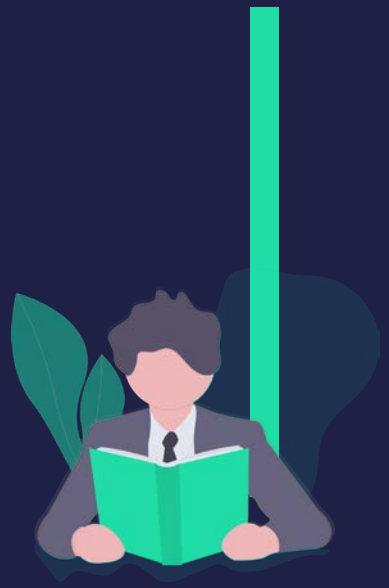


Les variables sont TYPÉES !

XWIKI NATIS

Type = Structure de données

XWIKI NATIS



Type = Structure de données

XWIKI SAS

XWIKI NATIS

Les types permettent à l'ordinateur d'identifier les actions possibles

XWIKI NATIS

Les types permettent à l'ordinateur d'identifier les actions possibles

Les types prennent un espace différent en RAM

XNATIS

Les types permettent à l'ordinateur d'identifier les actions possibles

Les types prennent un espace différent en RAM

XWIKI SAS

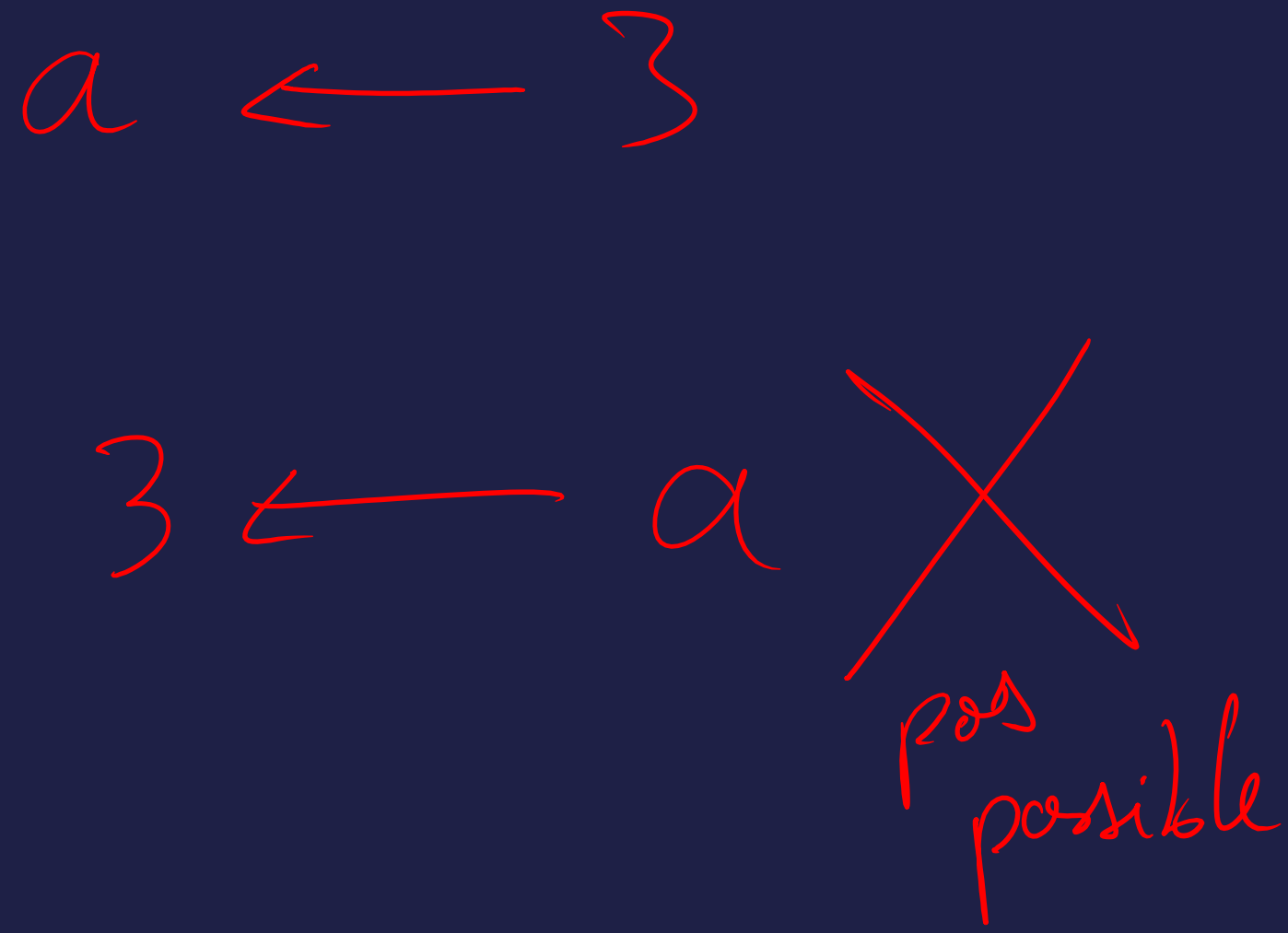
XNATIS

2. Instructions de base

XWIKI NATIS

Affectation

XWIKI NATIS



XWIKI NATIS

```
taille <- 34
```

XWIKI NATIS

```
taille <- 34
```

```
toto <- [23, 34, 32, 3]
```

XEBIA NATIS

```
taille <- 34
```

```
toto <- [23, 34, 32, 3]
```

```
resultat <- 'Petit'
```

XEBIA NATIS

```
taille <- 34
```

```
toto <- [23, 34, 32, 3]
```

```
resultat <- 'Petit'
```

```
yop <- Vrai
```

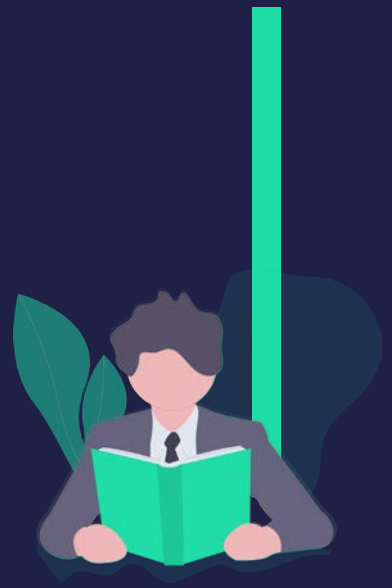
XNATIS

concaténation : string , string ⟶ string

Structure de données	Actions possibles
Nombre	Addition Soustraction Division Multiplication
Chaîne de caractères	Concaténation
Boolean	Et Ou Non
Tableau	Adressage (position) ~~Ajout~~ ~~Suppression~~
Dictionnaire	Adressage (clé) Ajout Suppression

XEBIA NATIS

Structure de données	Actions possibles
Nombre	Addition Soustraction Division Multiplication
Chaîne de caractères	Concaténation
Boolean	Et Ou Non
Tableau	Adressage (position) ~~Ajout~~ ~~Suppression~~
Dictionnaire	Adressage (clé) Ajout Suppression



XWIKI SAS

XWIKI SAS

XWIKI NATIS

3. Blocs

XWIKI NATIS

```
@DebutBloc
resultat <- 'Grand’
taille <- 34
@FinBloc
```

XWIKI NATIS

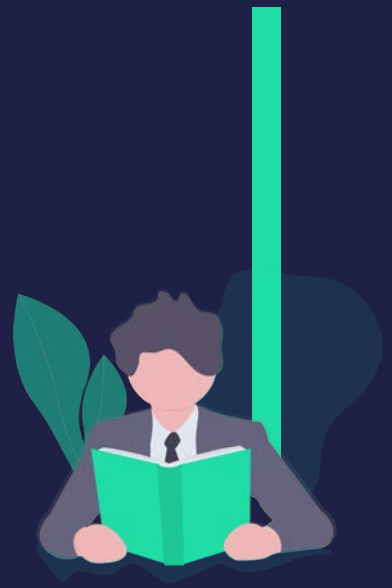
Un bloc permet de rassembler des instructions

XWIKI NATIS

Les variables définies dans un bloc meurent à la fin du bloc

XEBIA NATIS

Les variables définies dans un bloc meurent à la fin du bloc

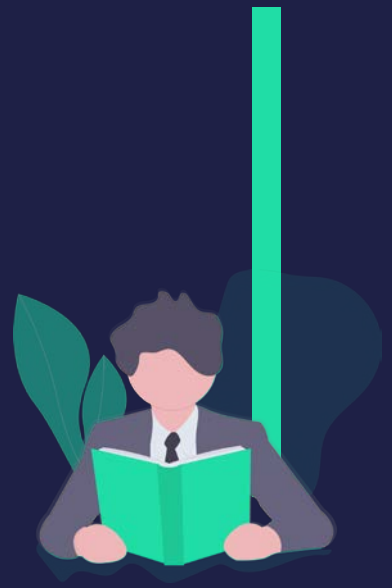


Portée (scope)

XWIKI NATIS

Un bloc est un ensemble d'instructions qui peuvent être conditionnés ou répétés

XNATIS



Un bloc est un ensemble d'instructions qui peuvent être conditionnés ou répétés

XWIKI SAS

XWIKI NATIS

4. Conditions

XWIKI NATIS

Une condition permet de conditionner l'exécution d'un bloc

XNATIS

```
taille <- 34

resultat <- 'Petit'
@Si taille >= 50
    @DebutBloc
    resultat <- 'Grand’
    @FinBloc
```

XWIKI NATIS

```
taille <- 34

resultat <- 'Petit'
@Si taille >= 50
    @DebutBloc
    resultat <- 'Grand’
    @FinBloc
```

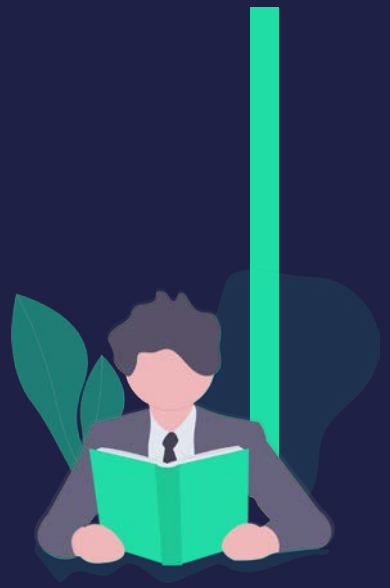
```
taille <- 34

resultat <- 'Petit'
@Si   @Non (taille < 50)
    @DebutBloc
    resultat <- 'Grand’
    @FinBloc
```

XWIKI NATIS

Une condition se base
sur 1 ou plusieurs
prédicats

XDNATIS

Une condition se base sur 1 ou plusieurs prédicats



XWIKI NATIS

La valeur logique d'un prédicat est toujours « Vrai » ou « Faux »

XNATIS

```
taille <- 34
forme <- 'Rectangle'

resultat <- 'Petit'
@Si taille >= 50 @Et forme = 'Rectangle’
    @DebutBloc
    resultat <- 'Grand’
    @FinBloc
```

XNATIS

```
taille <- 34
forme <- 'Rectangle'

resultat <- 'Petit'
@Si taille >= 50 @Ou forme = 'Rectangle’
    @DebutBloc
    resultat <- 'Grand’
    @FinBloc
```

XWIKI NATIS

```
taille <- 34
forme <- 'Rectangle'

@Si taille >= 50 @Ou forme = 'Rectangle’
    @DebutBloc
    resultat <- 'Grand’
    @FinBloc
@Sinon
    @DebutBloc
    resultat <- ’Petit’
    @FinBloc
```

XWIKI NATIS

Opérateurs binaires sur les prédicats

ET = « et en même temps … »

OU = « ou soit … »

XWIKI NATIS

Opérateurs binaires

a ET b

c OU d

XWIKI NATIS

Opérateurs unaires sur les prédicats

NON = « ne pas … » ou « contraire »

XWIKI NATIS

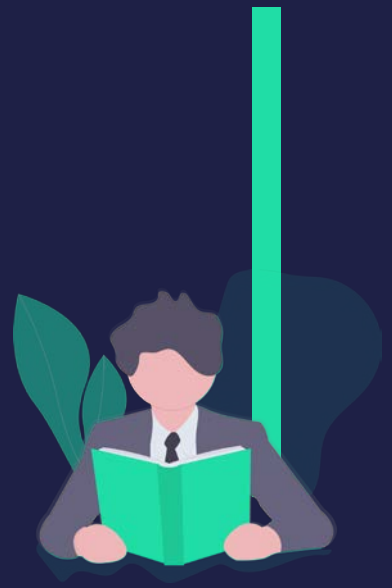
Opérateurs unaires sur les prédicats

NON (a)

XNATIS

Table de vérité

Les tables de vérité présentent tous les résultats possibles d'une opération logique



a	b	a ET b
Faux	Faux	Faux
Faux	Vrai	Faux
Vrai	Faux	Faux
Vrai	Vrai	Vrai

a	b	a OU b
Faux	Faux	Faux
Faux	Vrai	Vrai
Vrai	Faux	Vrai
Vrai	Vrai	Vrai

XWIKI SAS

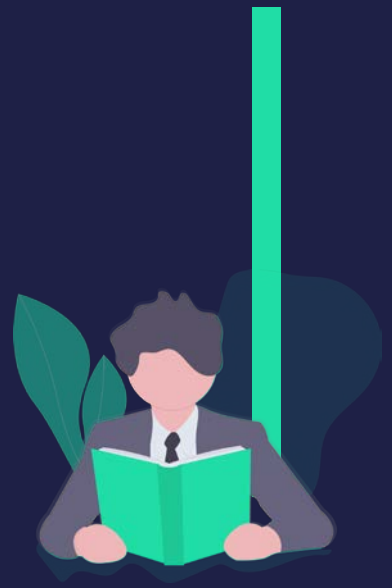
XDNATIS

Loi De Morgan

La loi De Morgan permet de « casser » un NON englobant un ET ou un OU

XEBIA NATIS

Loi De Morgan



NON (a ET b) = NON (a) OU NON (b)
NON (a OU b) = NON (a) ET NON (b)

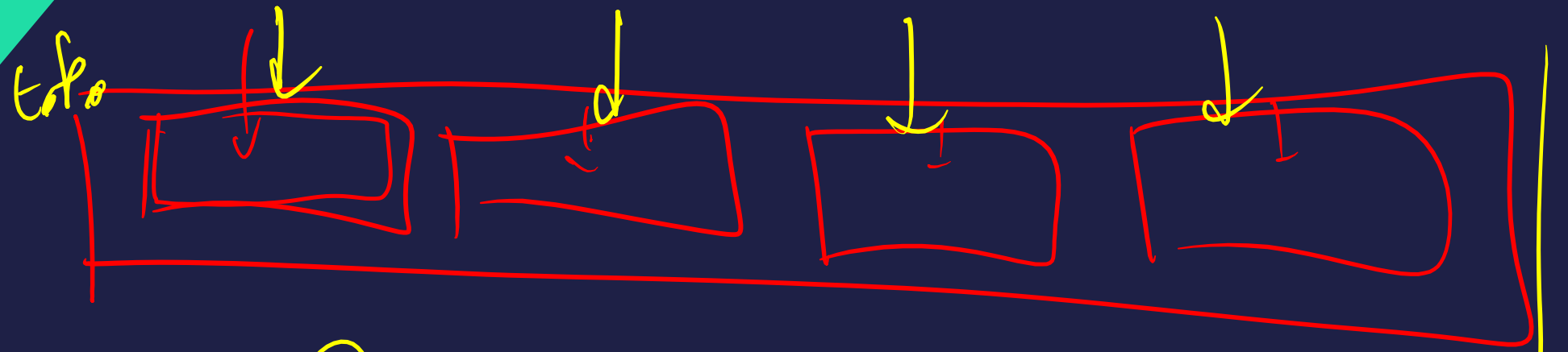
XWIKI SAS

XNATIS

tab [1] = valeur de l'élément à l'indice 1 du tab

tab [maison] = valeur de l'élément à l'indice maison du tab

XKNATIS



→ @ Pour Chaque element @ Dans tab

@ Debut Bloc

// 1 seul élément

@ Fin Bloc

XWIKI NATIS

5. Boucles

XDNATIS

Les boucles

Les boucles permettent de répéter un bloc d'instructions

XWIKI NATIS

Il y a 3 types de boucles pour répéter un bloc

1. @PourChaque
2. @Pour @De @A
3. @TantQue ou Boucle + @Stop

foreach
for
while

XEBIA NATIS

1. Il faut s'arrêter à la fin du tableau

```
tab <- [23, 43, 32, 4, 3]

@PourChaque element @Dans tab
    @DebutBloc
    Afficher element
    @FinBloc
```

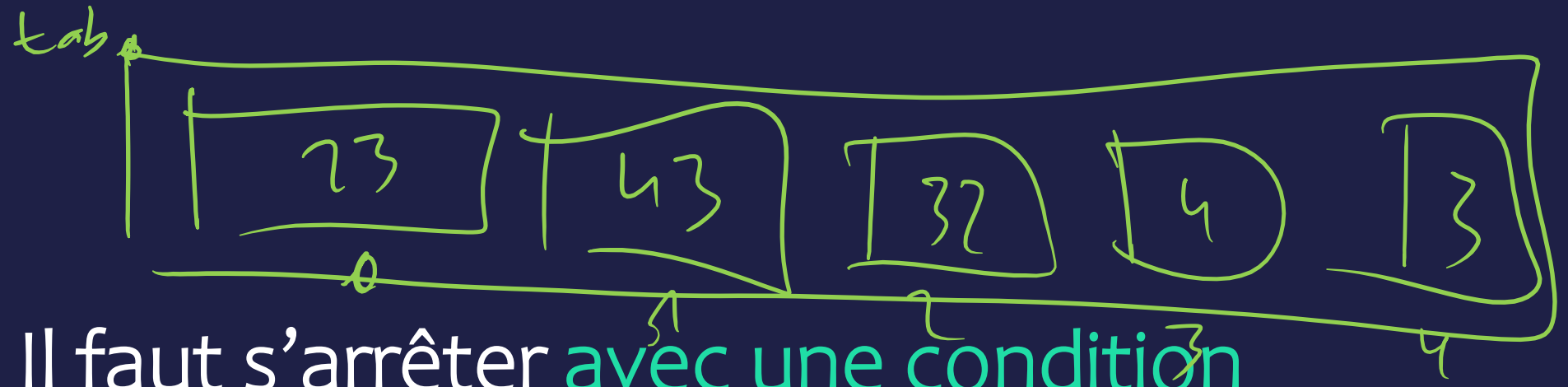
XNATIS

2. Il faut s'arrêter avec un nombre maximal

```
tab <- [23, 43, 32, 4, 3]

@Pour i @De 0 @A 4
    @DebutBloc
    Afficher tab[i]
    @FinBloc
```

(exclu)

XEBIANATIS



3. Il faut s'arrêter avec une condition


```
tab <- [23, 43, 32, 4, 3]

position <- 0
@TantQue tab[position] < 30
    @DebutBloc
    position <- position + 1
    @FinBloc
Afficher position
```

1

```
tab <- [23, 43, 32, 4, 3]

position <- 0
@PourChaque element @Dans tab
    @DebutBloc
    @Si element >= 30
        @DebutBloc
        Afficher position
        @Stop
        @FinBloc
    position <- position + 1
    @FinBloc
```

XWIKI SAS

XWIKI SAS

XWIKI SAS

XWIKI SAS