

Python - « IF »

I. Les tests

En programmation, et donc en Python, il est possible d'effectuer des tests sur des variables (ou autres).

1. Complétez les explications suivantes :

a. Le résultat d'un test est un booléen, c'est à dire qu'il vaut soit `True`, lorsque le test est, soit `False` lorsqu'il est

b. Python dispose de six opérateurs relationnels de base :

Python	<code>==</code>	<code>!=</code>	<code><</code>	<code>></code>	<code><=</code>	<code>>=</code>
Scratch						

c. Python dispose de trois opérateurs logiques :

Python	<code>and</code> ou <code>&</code>	<code>or</code> ou <code> </code>	<code>not</code>
Scratch			

I. Les tests

En programmation, et donc en Python, il est possible d'effectuer des tests sur des variables (ou autres).

1. Complétez les explications suivantes :

a. Le résultat d'un test est un booléen, c'est à dire qu'il vaut soit `True`, lorsque le test est **vrai**, soit `False` lorsqu'il est **Faux**.

b. Python dispose de six opérateurs relationnels de base :

Python	<code>==</code>	<code>!=</code>	<code><</code>	<code>></code>	<code><=</code>	<code>>=</code>
Scratch						

c. Python dispose de trois opérateurs logiques :

Python	<code>and</code> ou <code>&</code>	<code>or</code> ou <code> </code>	<code>not</code>
Scratch			

I. Les tests

En programmation, et donc en Python, il est possible d'effectuer des tests sur des variables (ou autres).

1. Complétez les explications suivantes :

a. Le résultat d'un test est un booléen, c'est à dire qu'il vaut soit `True`, lorsque le test est **vrai**, soit `False` lorsqu'il est **Faux**.

b. Python dispose de six opérateurs relationnels de base :

	égal à	Différent de	inférieur à	supérieur à	inférieur ou égal à.	Supérieur ou égal à.
Python	<code>==</code>	<code>!=</code>	<code><</code>	<code>></code>	<code><=</code>	<code>>=</code>
Scratch						

c. Python dispose de trois opérateurs logiques :

	ET	OU	NON
Python	<code>and</code> ou <code>&</code>	<code>or</code> ou <code> </code>	<code>not</code>
Scratch			

2. Dans la console de votre IDE tapez les lignes suivantes et notez les résultats obtenus.

```
>>> a=10
>>> b=5
>>> c=10-5
>>> a==b
>>> a==c
>>> a!=b
>>> a!=c
>>> a<b
>>> a<c
>>> a>b
>>> a<=b
>>> a<=c
>>> a>=b
```

3. Que doit-on saisir pour tester en une seule ligne si un nombre `n` appartient à l'intervalle `]1;4]` ?
4. Écrivez dans l'éditeur de votre IDE écrivez une fonction `dedans(n)` renvoyant `True` si `n` appartient à l'intervalle `]1;4]` et `False` sinon ?
5. Testez votre fonction directement dans la console, puis, si elle semble se comporter comme souhaité, appelez-moi pour valider ce premier exercice.

2. Dans la console de votre IDE tapez les lignes suivantes et notez les résultats obtenus.

```
>>> a=10
>>> b=5
>>> c=10-5
>>> a==b
>>> a==c
>>> a!=b
>>> a!=c
>>> a<b
>>> a<c
>>> a>b
>>> a<=b
>>> a<=c
>>> a>=b

>>> a = 10
>>> b = 5
>>> c = 10 - 5
>>> a == b # False
>>> a == c # False
>>> a != b # True
>>> a != c # True
>>> a < b # False
>>> a < c # False
>>> a > b # True
>>> a <= b # False
>>> a <= c # False
>>> a >= b # True
```

3. Que doit-on saisir pour tester en une seule ligne si un nombre `n` appartient à l'intervalle `]1;4]`?
4. Écrivez dans l'éditeur de votre IDE écrivez une fonction `dedans(n)` renvoyant `True` si `n` appartient à l'intervalle `]1;4]` et `False` sinon?
5. Testez votre fonction directement dans la console, puis, si elle semble se comporter comme souhaité, appelez-moi pour valider ce premier exercice.

3. Que doit-on saisir pour tester en une seule ligne si un nombre `n` appartient à l'intervalle `]1;4]`?
4. Écrivez dans l'éditeur de votre IDE écrivez une fonction `dedans(n)` renvoyant `True` si `n` appartient à l'intervalle `]1;4]` et `False` sinon?
5. Testez votre fonction directement dans la console, puis, si elle semble se comporter comme souhaité, appelez-moi pour valider ce premier exercice.

3. Que doit-on saisir pour tester en une seule ligne si un nombre `n` appartient à l'intervalle `]1;4]`?

$$1 < n \leq 4$$

4. Écrivez dans l'éditeur de votre IDE écrivez une fonction `dedans(n)` renvoyant `True` si `n` appartient à l'intervalle `]1;4]` et `False` sinon?

5. Testez votre fonction directement dans la console, puis, si elle semble se comporter comme souhaité, appelez-moi pour valider ce premier exercice.

3. Que doit-on saisir pour tester en une seule ligne si un nombre `n` appartient à l'intervalle `]1;4]`?

$$1 < n \leq 4$$

4. Écrivez dans l'éditeur de votre IDE écrivez une fonction `dedans(n)` renvoyant `True` si `n` appartient à l'intervalle `]1;4]` et `False` sinon?

```
def dedans(n):  
    return 1 < n <= 4
```

5. Testez votre fonction directement dans la console, puis, si elle semble se comporter comme souhaité, appelez-moi pour valider ce premier exercice.