

Session tuples et affectations multiples

```
>>> [0, 2] # une liste
[0, 2]
>>> (0, 2) # un tuple
(0, 2)
>>> len((0, 2)) # un tuple a une longueur
2
>>> t0 = () # le tuple vide (totalement inutile)
>>> t1 = (42,) # besoin d'une virgule pour un tuple
# de longueur 1
>>> t1.append(2) # on ne peut pas « append » à un tuple !
Traceback (most recent call last):
AttributeError: 'tuple' object has no attribute 'append'

>>> (1, 2, 3)[2] # on lit un élément comme dans une liste
3
>>> t3 = 1, 2, 3 # les parenthèses sont facultatives
>>> type(t3)
<class 'tuple'>
>>> t3[2]
3
>>> (a, b) = (42, 666) # affectation multiple
>>> a # avec ou sans parenthèses !!!
42 # voir plus bas
>>> b
666
>>> [[a, b], c] = ((111, 222), 333) # idem mais avec
>>> a # une structure
111 # plus complexe
>>> b
222
>>> c
333
>>> def deux_valeurs(): # fonction qui retourne un tuple
... return 666, 42 # (meilleure façon de retourner
... # plusieurs valeurs)
>>> a, b = deux_valeurs()
>>> a
666
>>> b
42
```

Session tuples et affectations multiples

```
>>> [0, 2] # une liste
[0, 2]
>>> (0, 2) # un tuple
(0, 2)
>>> len((0, 2)) # un tuple a une longueur
2
>>> t0 = () # le tuple vide (totalement inutile)
>>> t1 = (42,) # besoin d'une virgule pour un tuple
# de longueur 1
>>> t1.append(2) # on ne peut pas « append » à un tuple !
Traceback (most recent call last):
AttributeError: 'tuple' object has no attribute 'append'

>>> (1, 2, 3)[2] # on lit un élément comme dans une liste
3
>>> t3 = 1, 2, 3 # les parenthèses sont facultatives
>>> type(t3)
<class 'tuple'>
>>> t3[2]
3
>>> (a, b) = (42, 666) # affectation multiple
>>> a # avec ou sans parenthèses !!!
42 # voir plus bas
>>> b
666
>>> [[a, b], c] = ((111, 222), 333) # idem mais avec
>>> a # une structure
111 # plus complexe
>>> b
222
>>> c
333
>>> def deux_valeurs(): # fonction qui retourne un tuple
... return 666, 42 # (meilleure façon de retourner
... # plusieurs valeurs)
>>> a, b = deux_valeurs()
>>> a
666
>>> b
42
```