Copying in Python

	a = [1, 2, 3, 4] --> Copy to b list		
	Aliasing	Shallow copy	Deep copying
syntax	b = a	b = copy(a) b = a[:]	b = deepcopy(a)
id(a) == id(b)	TRUE	FALSE	FALSE
id(a[i]) == id(b[i])	TRUE	TRUE	TRUE for immutable obj FALSE for mutable obj
change of element	Reflects	Reflects	Does not reflect at all

Shallow Copy

Deep Copy

By Vinodhini Sadasivam Gnanavadivel
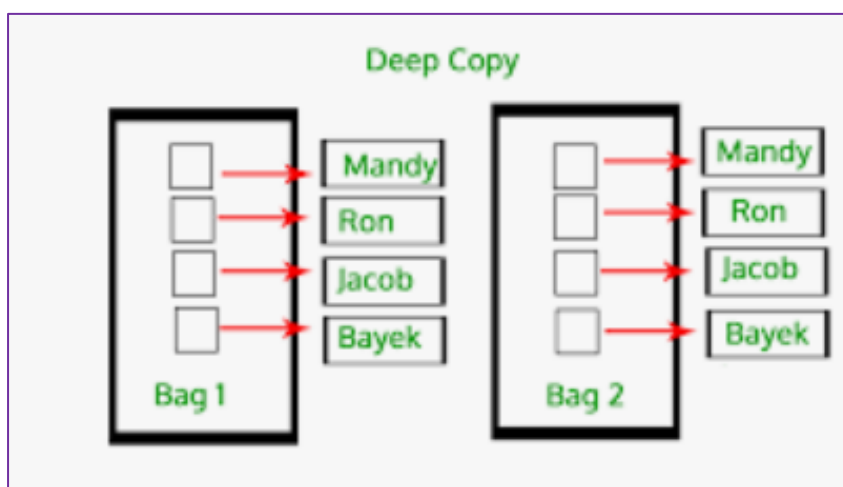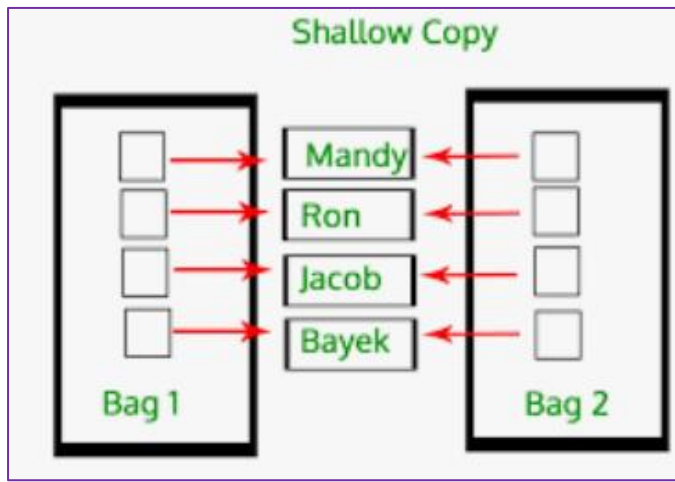
Aliasing method - Code:

Base address and the element address of old and copied lists (object's residence/reference) are same.

```python
lst1 = ["Land", "of", "Wonders", "is", "awaiting"]

# aliasing

lst2 = lst1

print("Aliasing--------------")
print("lst1: ", lst1, "lst2: ", lst2, sep = '\n')
print("Base address: ",id(lst2), id(lst1))

for i in range(len(lst1)):
    print("\"",lst1[i],"\""," of lst1 is residing at: ",id(lst1[i]))
    print("\"",lst2[i],"\""," of lst2 is residing at: ",id(lst2[i]))
    print()
```
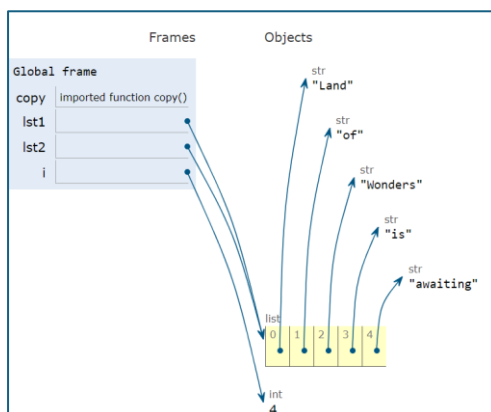
Visualization:

Output: Illustrated using id()

By Vinodhini Sadasivam Gnanavadivel

Shallow copying – code:

Base address of old and copied lists (object's residence/reference) are different.

The elements' address of old and copied lists (object's residence/reference) are same.

```python
from copy import copy
from copy import deepcopy

lst3 = ["Land", "of", "Wonders", "is", "awaiting", (1, 2, 3), [60, 70]]


#shallow copying

lst4 = copy(lst3)
print("\n Shallow copying--------------")
print("\n lst1: ", lst3, "lst4: ", lst4, sep = '\n')
print("Base address: ",id(lst4), id(lst3))
for k in range(len(lst3)):
    print("\"",lst3[k],"\""," of lst3 is residing at: ",id(lst3[k]))
    print("\"",lst4[k],"\""," of lst4 is residing at: ",id(lst4[k]))
    print()
```

Visulaization:
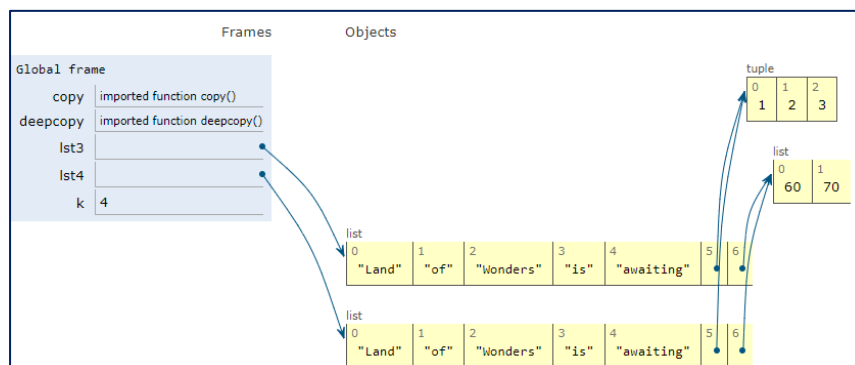
Output:

```
 Shallow copying--------------

 lst1:
['Land', 'of', 'Wonders', 'is', 'awaiting', (1, 2, 3), [60, 70]]
lst4:
['Land', 'of', 'Wonders', 'is', 'awaiting', (1, 2, 3), [60, 70]]
Base address:  140379894491520 140379894515392
" Land "  of lst3 is residing at:  140379894866352
" Land "  of lst4 is residing at:  140379894866352

" of "  of lst3 is residing at:  140379894868784
" of "  of lst4 is residing at:  140379894868784

" Wonders "  of lst3 is residing at:  140379894868656
" Wonders "  of lst4 is residing at:  140379894868656

" is "  of lst3 is residing at:  140379758423088
" is "  of lst4 is residing at:  140379758423088

" awaiting "  of lst3 is residing at:  140379894811056
" awaiting "  of lst4 is residing at:  140379894811056

" (1, 2, 3) "  of lst3 is residing at:  140379733248576
" (1, 2, 3) "  of lst4 is residing at:  140379733248576

" [60, 70] "  of lst3 is residing at:  140379894490688
" [60, 70] "  of lst4 is residing at:  140379894490688
```

By Vinodhini Sadasivam Gnanavadivel

Deep copying – code:

Base address of old and copied lists (object's residence/reference) are different.

The elements' address of old and copied lists (object's residence/reference) are same for immutable objects and different for mutable objects.

```python
from copy import copy
from copy import deepcopy

lst3 = ["Land", "of", "Wonders", "is", "awaiting", (1, 2, 3), [60, 70]]


#deep copying

lst4 = deepcopy(lst3)
print("\n deep copying--------------")
print("\n lst1: ", lst3, "lst4: ", lst4, sep = '\n')
print("Base address: ",id(lst4), id(lst3))
for k in range(len(lst3)):
    print("\"",lst3[k],"\""," of lst3 is residing at: ",id(lst3[k]))
    print("\"",lst4[k],"\""," of lst4 is residing at: ",id(lst4[k]))
    print()
```

Output:

```
deep copying--------------

lst1:
['Land', 'of', 'Wonders', 'is', 'awaiting', (1, 2, 3), [60, 70]]
lst4:
['Land', 'of', 'Wonders', 'is', 'awaiting', (1, 2, 3), [60, 70]]
Base address:  140505493285248 140505493309120
" Land "  of lst3 is residing at:  140505356921520
" Land "  of lst4 is residing at:  140505356921520

" of "  of lst3 is residing at:  140505356921456
" of "  of lst4 is residing at:  140505356921456

" Wonders "  of lst3 is residing at:  140505493604848
" Wonders "  of lst4 is residing at:  140505493604848

" is "  of lst3 is residing at:  140505357216688
" is "  of lst4 is residing at:  140505357216688

" awaiting "  of lst3 is residing at:  140505493603888
" awaiting "  of lst4 is residing at:  140505493603888

" (1, 2, 3) "  of lst3 is residing at:  140505332028224
" (1, 2, 3) "  of lst4 is residing at:  140505332028224

" [60, 70] "  of lst3 is residing at:  140505493284416
" [60, 70] "  of lst4 is residing at:  140505317981312
```

Visualization:

By Vinodhini Sadasivam Gnanavadivel