



Tras ver cómo trata Python el paso por valor y por referencia de parámetros y su relación con objetos mutables e inmutables ejecuta los fragmentos de código dados (ver aula virtual):

1. Depúralos para ver qué resultado se obtiene en cada uno de los casos.
2. Argumenta el porqué de cada resultado.

Realiza la tarea indicada para los ejercicios del 1 al 7 (Los de clases los comentaremos entre todos)

### Entrega:

- Documento PDF con 2 pantallazos para cada uno de los puntos en los que se vea en depuración el valor de las variables involucradas en el ejemplo.
- Los pantallazos deben tener recuadrada la variable y su resultado
- Nombre del PDF: **Apellidos\_Nombre\_PIA\_UT2\_EJ4\_Mutabilidad.pdf**.
- **No se admitirá ningún fichero que no tenga el nombre solicitado.**
- **TODOS los pantallazos deben incluir el nombre del estudiante con sesión iniciada en las aulas virtuales o educamos.**
- Fecha de entrega: Se indica en el aula virtual
- Duración estimada: 2 horas.

### Ejercicios:

1.

```
#1
def modificar_entero(x):
    x += 5
    return x

a = 5
resultado = modificar_entero(a)
print(a) # ¿Qué valor se imprime?
print(resultado) # ¿Qué valor se imprime?
```

2.

El primer print es 5 el segundo 10



```
#2
def agregar_elemento(lista, elemento):
    lista.append(elemento)
    return lista

numeros = [1, 2, 3]
agregar_elemento(numeros, 4)
print(numeros) # ¿Qué valor se imprime?
```

Imprime la lista de números 1 2 3 4

3.

```
#3
def concatenar_y_agregar(s, lista, elemento):
    s += " mundo"
    lista.append(elemento)
    return s, lista

texto = "Hola"
numeros = [1, 2, 3]
resultado_texto, resultado_lista = concatenar_y_agregar(texto, numeros, 4)
print(texto) # ¿Qué valor se imprime?
print(numeros) # ¿Qué valor se imprime?
print(resultado_texto) # ¿Qué valor se imprime?
print(resultado_lista) # ¿Qué valor se imprime?
```

Primero = hola

Segundo = 1,2,3

Tercero = Hola mundo

Cuarto 1,2,3,4



4.

```
#4
class MiClase:
    def __init__(self, valor):
        self.valor = valor

    def modificar_objeto(objeto):
        objeto.valor += 10
        return objeto

mi_objeto = MiClase(5)
modificar_objeto(mi_objeto)
print(mi_objeto.valor) # ¿Qué valor se imprime?
```

Imprimiría 15

5.

```
#5
elemento_global = 5

def agregar_elemento_global(lista):
    global elemento_global
    lista.append(elemento_global)
    elemento_global += 1
    return lista

numeros = [1, 2, 3]
agregar_elemento_global(numeros)
print(numeros) # ¿Qué valor se imprime?
print(elemento_global) # ¿Qué valor se imprime?
```

Primero 1,2,3,5

Segundo 6

6.

```
#6
def modificar_mezcla(x, lista):
    x *= 2
    #funcion que cambia el orden de los elementos de la lista
    lista.reverse()
    return x, lista

a = 10
numeros = [4, 5, 6]
resultado_a, resultado_lista = modificar_mezcla(a, numeros)
print(a) # ¿Qué valor se imprime?
print(numeros) # ¿Qué valor se imprime?
print(resultado_a) # ¿Qué valor se imprime?
print(resultado_lista) # ¿Qué valor se imprime?
```

Primero 10

Segundo 4,5,6

Tercero 20

Cuarto 6,5,4



7.

```
#7
def modificar_por_indice(lista, indice, valor):
    if indice < len(lista):
        lista[indice] = valor
    return lista

numeros = [1, 2, 3, 4]
modificar_por_indice(numeros, 2, 99)
print(numeros) # ¿Qué valor se imprime?
```

Imprime 1,2,99,4

8.

```
#8
class Contenedor:
    def __init__(self, contenido):
        self.contenido = contenido

def cambiar_contenido(contenedor, nuevo_contenido):
    contenedor.contenido = nuevo_contenido
    nuevo_contenido += " Contenido"
    return nuevo_contenido

mi_contenedor = Contenedor("Antiguo")
nuevo_contenido = cambiar_contenido(mi_contenedor, "Nuevo")
print(mi_contenedor.contenido) # ¿Qué valor se imprime?
print(nuevo_contenido) # ¿Qué valor se imprime?
```

Primero nuevo

Segundo contenido nuevo

9.

```
#9
class Contador:
    valor = 10 # Atributo de clase

def incrementar_contador():
    global Contador
    Contador.valor += 1
    return Contador.valor

incrementar_contador()
print(Contador.valor) # ¿Qué valor se imprime?
```

Imrpime 11