## Ejercicios faltantes de los resumenes

### Ejercicio 1/1.1:

- `mint.make_money()`

### Ejercicio 2/1.1:

- `make_money()`

### Ejercicio 3/1.1:

- `from mint import make_money as mint_money`

### Ejercicio 4/1.1:

- `make_money()`

### Ejercicio 1/1.2:

- `True`

### Ejercicio 2/1.2:

- Se generará la misma secuencia de números pseudoaleatorios desde `random`

### Ejercicio 3/1.2:

- Función `processor()`

### Ejercicio 4/1.2:

- `3`

### Ejercicio 1/1.3:

```
if __name__ == "__main__":
    print("Este código no es un script ordinario")
    sys.exit()
```

### Ejercicio 2/1.3:

```
import sys
sys.path.append("D:\\Python\\Project\\Modules")
```

### Ejercicio 3/1.3:

```
import abc.def.mymodule
```

### Ejercicio 1/1.4:

- El nombre "The Cheese Shop" proviene de un famoso sketch de los Monty Python titulado "Cheese Shop"

### Ejercicio 2/1.4:

- `pip` o `pip3` dependiendo de la versión de Python que estés utilizando. `pip` se asocia con Python 2, mientras que `pip3` se asocia con Python 3

### Ejercicio 3/1.4:

- `pip --version`

### Ejercicio 4/1.4:

- Consultar con el administrador del sistema

### Ejercicio 1/2:

- BOM significa "Byte Order Mark" y se utiliza para indicar el orden de bytes en archivos, principalmente en UTF-16 y UTF-32

### Ejercicio 2/2:

- Si

### Ejercicio 1/2.2:

- `1`

### Ejercicio 2/2.2:

- `['t', 'e', 'r']`

### Ejercicio 3/2.2:

- `bcd`

### Ejercicio 1/2.3:

- `ABC123xyx`

### Ejercicio 2/2.3:

- `de`

### Ejercicio 3/2.3:

- ¿Dónde*están*las*nevadas?

### Ejercicio 4/2.3:

- Es difícil o posible

### Ejercicio 1/2.4:

- `1` y `4`

### Ejercicio 2/2.4:

- `de`

### Ejercicio 3/2.4:

- `False`

### Ejercicio 1/2.6:

```
 Tratemos de hacer esto
Hemos fallado
Hemos terminado
```

### Ejercicio 2/2.6:

- `cero`

### Ejercicio 1/2.7:

- `cero`

### Ejercicio 2/2.7:

- `arit`

### Ejercicio 3/2.7:

- `algo`

### Ejercicio 1/2.8:

- `KeyboardInterrupt`

### Ejercicio 2/2.8:

- `BaseException`

### Ejercicio 3/2.8:

- `OverflowError`

### Ejercicio 1/3.1:

- Serpiente, reptil, vertebrado.

### Ejercicio 2/3.1:

- Pitón india, Pitón de Roca Sfricana.

### Ejercicio 3/3.1:

- No, es una palabra reservada.

### Ejercicio 1/3.2:

```
class Python(Snakes):
```

### Ejercicio 2/3.2:

- Falta el constructor `__init__()`

### Ejercicio 3/3.2:

```
class Snakes:
    def __init__(self):
        self.venomous = True
```

### Ejercicio 1/3.3:

- Variables de clase: `population` y `victims`. Variable de instancia: `length_ft`. Variable de instancia privada: `__venomous`.

### Ejercicio 2/3.3:

- `version_2-_Python__venomous = not version_2-_Python__venomous`

### Ejercicio 3/3.3:

- `hasattr(version_2, 'constrictor')`

### Ejercicio 1/3.4:

```
class Snake:
    def __init__(self):
        self.victims = 0

    def increment(self):
        self.victims += 1
```

### Ejercicio 2/3.4:

```
class Snake:
    def __init__(self, victims=0):
        self.victims = victims
```

### Ejercicio 3/3.4:

- Python es una Snake Snake puede ser una Python

### Ejercicio 1/3.5:

```
Collie dice: ¡Guau! ¡No huyas, corderito!
Dobermann dice: ¡Guau! ¡Quédese donde está, intruso!
```

### Ejercicio 2/3.5:

- `(True, False)` `(False, True)`

### Ejercicio 3/3.5:

- `True False` `2`

### Ejercicio 4/3.5:

```
class LowlandDog(SheepDog):
    def __str__(self):
        return "¡Guau! ¡No me gustan las montañas!"
```

### Ejercicio 1/3.6:

- `3.0` `ok`

### Ejercicio 2/3.6:

- `inf` `fin`

### Ejercicio 3/3.6:

- `('Advertencia enemiga', 'Alerta roja', 'Alta disponibilidad')`

### Ejercicio 1/4.1:

- `a e i o u y`

### Ejercicio 2/4.1:

- `list(map(lambda x: x * 2 - 1, any_list))`

### **Ejercicio

3/4.1:**

- `And*Now*for*Something*Completely*Different`

### Ejercicio 1/4.2:

- `w`

### Ejercicio 2/4.2:

- `Permiso denegado`

### Ejercicio 3/4.2:

- `ausente`

### Ejercicio 1/4.3:

- `[]`

### Ejercicio 2/4.3:

- El siguiente código pretende imprimir en la consola todos los caracteres del archivo llamado "file"

### Ejercicio 3/4.3:

- `image = bytearray(stream.read())`

### Ejercicio 1/4.4:

- `posix`

### Ejercicio 2/4.4:

- `['hello']`

### Ejercicio 1/4.5:

- `14:39:00`

### Ejercicio 2/4.5:

- `1. day, 0:00:00`

### Ejercicio 4/4.6:

- `M T W T F S S`

### Ejercicio 4/4.6:

- `0 1 2 3 4 5 6`