

## Comprensión de Colecciones

En muchas aplicaciones es habitual aplicar una función o realizar una operación con los elementos de una colección (lista, tupla o diccionario) y obtener una nueva colección de elementos transformados. Aunque esto se puede hacer recorriendo la secuencia con un bucle iterativo, y en programación funcional mediante la función `map`, Python incorpora un mecanismo muy potente que permite esto mismo de manera más simple.

### 1. Comprensión de listas

[*expresion* **for** *variable* **in** *lista* **if** *condicion*]

Esta instrucción genera la lista cuyos elementos son el resultado de evaluar la expresión *expresion*, para cada valor que toma la variable **variable**, donde **variable** toma todos los valores de la lista **lista** que cumplen la condición **condición**.

---

**Ejercicio de Inducción:** Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> [x ** 2 for x in range(10)]
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
>>> [x for x in range(10) if x % 2 == 0]
[0, 2, 4, 6, 8]
>>> [x ** 2 for x in range(10) if x % 2 == 0]
[0, 4, 16, 36, 64]
>>> notas = {'Carmen':5, 'Antonio':4, 'Juan':8, 'Mónica':9, 'María': 6, 'Pablo':3}
>>> [nombre for (nombre, nota) in notas.items() if nota >= 5]
['Carmen', 'Juan', 'Mónica', 'María']
```

### 2. Comprensión de diccionarios

*expresion-clave:expresion-valor* **for** *variables* **in** *lista* **if** *condición*

Esta instrucción genera el diccionario formado por los pares cuyas claves son el resultado de evaluar la expresión *expresion-clave* y cuyos valores son el resultado de evaluar la expresión *expresion-valor*, para cada valor que toma la variable **variable**, donde **variable** toma todos los valores de la lista **lista** que cumplen la condición **condición**.

---

**Ejercicio de Inducción:** Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> {palabra:len(palabra) for palabra in ['I', 'love', 'Python']}
{'I': 1, 'love': 4, 'Python': 6}
>>> notas = {'Carmen':5, 'Antonio':4, 'Juan':8, 'Mónica':9, 'María': 6, 'Pablo':3}
>>> {nombre: nota + 1 for (nombre, nota) in notas.items() if nota >= 5}
{'Carmen': 6, 'Juan': 9, 'Mónica': 10, 'María': 7}
```