

Ejercicios Prácticos: Diccionarios en Python

Nivel 1: Básico (Sintaxis, Acceso y Modificación)

Objetivo: Familiarizarse con la creación de diccionarios, la estructura `{clave: valor}` y las operaciones CRUD básicas.

1. **El Diccionario Personal** Crea un diccionario llamado `mi_perfil` que contenga las siguientes claves: `'nombre'`, `'edad'`, `'ciudad'` y `'hobby'`. Asigna tus propios datos y luego imprime el diccionario completo en pantalla.
2. **Traductor Simple** Crea un diccionario con 5 palabras en español como claves y sus traducciones en inglés como valores. El programa debe pedir al usuario una palabra en español y mostrar su traducción.
3. **Acceso Seguro** Usando el diccionario del ejercicio anterior, intenta buscar una palabra que no exista en tu lista. Utiliza el método `.get()` para que el programa imprima "Palabra no encontrada" en lugar de detenerse con un error.
4. **Actualización de Datos** Dado el siguiente diccionario: `coche = {'marca': 'Ford', 'modelo': 'Mustang', 'año': 1964}`. Escribe el código para cambiar el valor de la clave `'año'` a 2024 y agrega una nueva clave llamada `'color'` con el valor `'rojo'`.
5. **Eliminación Básica** Crea un diccionario que contenga al menos 3 frutas y sus precios. Elimina una de las frutas usando la palabra clave `del` o el método `.pop()` e imprime el precio de la fruta que acabas de borrar.

Nivel 2: Intermedio (Iteración y Lógica)

Objetivo: Practicar el uso de bucles for, métodos de vista (`.items()`, `.values()`, `.keys()`) y lógica condicional.

6. **Formato de Salida** Crea un diccionario con 3 países y sus capitales. Recorre el diccionario usando un bucle for y el método `.items()` para imprimir, por cada par, una frase completa como: *"La capital de [País] es [Capital]"*.
7. **Suma de Valores** Dado el diccionario de gastos: `gastos = {'Luz': 1500, 'Agua': 500, 'Internet': 1200}`. Utiliza el método `.values()` dentro de un bucle o con la función `sum()` para calcular e imprimir el total de gastos mensuales.
8. **Filtro de Aprobados** Dado el diccionario de calificaciones: `notas = {'Ana': 8, 'Luis': 4, 'Maria': 9, 'Pedro': 5}`. Escribe un código que cree un **nuevo** diccionario llamado `aprobados` que contenga solo a los estudiantes con nota mayor o igual a 6.
9. **Contador de Frecuencia** Dada la lista de compras: `lista = ['pan', 'leche', 'pan', 'huevos', 'leche', 'pan']`. Escribe un programa que use un diccionario para contar cuántas veces aparece cada producto en la lista (Ejemplo: `{ 'pan': 3, ... }`).
10. **Diccionario desde Listas** Dadas dos listas, una de nombres = `['Ana', 'Juan']` y otra de edades = `[25, 30]`, escribe un código que las combine para crear un único diccionario `{ 'Ana': 25, 'Juan': 30 }`.

Nivel 3: Avanzado (Métodos Específicos y Manipulación)

Objetivo: Dominar métodos avanzados como `update`, `setdefault`, `popitem`, `fromkeys` y el manejo de referencias con `copy`.

11. **Fusión de Inventarios** Tienes dos diccionarios: `tienda_A = {'manzana': 10, 'pera': 5}` y `tienda_B = {'pera': 8, 'uva': 12}`. Usa el método `.update()` para unificar el inventario de la tienda B dentro de la tienda A e imprime el resultado final. Explica qué sucedió con el valor de la 'pera'.
12. **Inicialización Masiva** Tienes una lista de nuevos usuarios: `usuarios = ['Alicia', 'Beto', 'Carla']`. Utiliza el método de clase `dict.fromkeys()` para crear rápidamente un diccionario donde cada usuario tenga una puntuación inicial de 0.
13. **Gestión de Configuraciones (Setdefault)** Estás simulando la configuración de un juego. Tienes un diccionario llamado `config`. Usa el método `.setdefault()` para verificar si existe la clave 'dificultad'. Si no existe, asígnala al valor 'Normal' y almacena el resultado en una variable.
14. **Pila de Tareas (Popitem)** Crea un diccionario con 3 tareas pendientes (ej. `'tarea1': 'limpiar', 'tarea2': 'estudiar'...`). Usa un bucle `while` y el método `.popitem()` para ir extrayendo y mostrando las tareas una por una hasta que el diccionario quede completamente vacío.
15. **Backup y Modificación**
 - Crea un diccionario con datos de un empleado.
 - Crea una copia de seguridad usando `.copy()`.
 - Elimina todos los datos del diccionario **original** usando `.clear()`.
 - Imprime ambos diccionarios para demostrar que la copia de seguridad conserva los datos originales intactos.