



Universidad Autónoma de Baja California

Facultad de Ingeniería, Arquitectura y Diseño

Ingeniero en Computación

Asignatura:

Lenguaje de Programación Python

Actividad 10:

Diccionarios

Brayan Arturo Rocha Meneses

Matricula:

371049

Ensenada Baja California 29 de Octubre del 2023

Introducción:

Los diccionarios son una estructura de datos fundamental en Python que permite almacenar y organizar datos de manera eficiente. A diferencia de las listas, los diccionarios no están indexados numéricamente, sino que utilizan claves para acceder a los valores asociados. Cada clave está vinculada a un valor específico, lo que facilita la recuperación de datos de manera rápida y sencilla.

Competencia:

El uso efectivo de diccionarios en Python es esencial para el desarrollo de aplicaciones y la manipulación de datos. Al comprender cómo funcionan los diccionarios y cómo se pueden utilizar para organizar información, los programadores pueden escribir código más eficiente y legible.

Fundamentos:

Creación de diccionarios

Los diccionarios se crean utilizando llaves {} y pares clave-valor separados por dos puntos :. Por ejemplo:

```
mi_diccionario = {"nombre": "Juan", "edad": 30, "ciudad": "Ejemplo"}
```

Acceso a elementos

Para acceder a un valor en un diccionario, se utiliza la clave correspondiente:

```
nombre = mi_diccionario["nombre"] # Acceder al valor asociado con la clave "nombre"
```

Modificación y adición de elementos

Puedes modificar el valor asociado a una clave existente o agregar nuevas claves-valor:

```
mi_diccionario["edad"] = 31 # Modificar el valor de la clave "edad"
```

```
mi_diccionario["ocupación"] = "Desarrollador" # Agregar una nueva clave-valor
```

Eliminación de elementos

Puedes eliminar un elemento de un diccionario utilizando la palabra clave del:

```
del mi_diccionario["ciudad"] # Eliminar la clave "ciudad" y su valor asociado
```

Métodos de diccionarios

Python proporciona una variedad de métodos para trabajar con diccionarios, como keys(), values(), items(), que permiten acceder a las claves, valores y pares clave-valor respectivamente.

Ejemplos

A continuación, se presentan algunos ejemplos de cómo se pueden utilizar los diccionarios en Python:

```
# Crear un diccionario vacío
```

```
mi_diccionario = {}
```

Agregar elementos al diccionario

```
mi_diccionario["nombre"] = "María"
```

```
mi_diccionario["edad"] = 25
```

Acceder a un valor

```
print("Nombre:", mi_diccionario["nombre"])
```

```
print("Edad:", mi_diccionario["edad"])
```

Modificar un valor

```
mi_diccionario["edad"] = 26
```

Eliminar un elemento

```
del mi_diccionario["edad"]
```

Verificar si una clave existe en el diccionario

```
if "nombre" in mi_diccionario:
```

```
    print("La clave 'nombre' existe en el diccionario")
```

Recorrer un diccionario

```
for clave, valor in mi_diccionario.items():
```

```
    print(clave, ":", valor)
```

Procedimiento:

REALIZA UN PROGRAMA QUE UTILICE UNA LISTA DE DICCIONARIOS

MENU

- 1.- Agregar (automatico)
- 2.- Agregar (manual)
- 3.- Imprimir lista
 - a)
 - b)
 - c)
- 4.- Buscar {ID}
- 5.- Ordenar
- 6.- Eliminar {ID}
- 7.- Borrar Toda la lista {ID}
- 0.- SALIR

NOTA: Los datos del diccionario son los Datos básicos de un trabajador de una fabrica.

NOTA 2: VALIDAR EL PROGRAMA 100%

Preguntar siempre si esta seguro eliminar, No ordenar si ya esta Ordenado, decir si lista vacía, no existe o mostrar si estala búsqueda, No se permiten ID repetidos

Conclusiones:

Los diccionarios son una herramienta poderosa en Python para organizar y acceder a datos de manera eficiente. Al utilizar claves en lugar de índices numéricos, se simplifica la búsqueda y recuperación de valores. Además, los métodos proporcionados por Python para trabajar con diccionarios facilitan la manipulación de datos en aplicaciones y scripts.

Es esencial comprender cómo crear, acceder, modificar y eliminar elementos en un diccionario, ya que esto contribuye a la legibilidad y eficiencia del código. Los diccionarios son ampliamente utilizados en Python y son una parte fundamental del repertorio de cualquier programador que trabaje en este lenguaje.

Anexo:

[https://github.com/BrayanArM98/Python/blob/main/RMBA PY ACT10.pdf](https://github.com/BrayanArM98/Python/blob/main/RMBA_PY_ACT10.pdf)