



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### Ingeniero en computación

### Ingeniero en Software y tecnologías emergentes

**Materia:** Programación en Python / Clave COM432

**Alumno:** Emiliano Nevarez Palma

**Matrícula:** 375295

**Maestro:** Pedro Núñez Yépiz

**Actividad No. : 10**

**Tema - Unidad :** Lista de diccionarios

**Ensenada Baja California a 20 de abril del 2023**



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 1. INTRODUCCIÓN

Creación de listas mediante diccionarios

### 2. COMPETENCIA

Conocer como crear listas apartir de diccionarios y trabajar con ello

### 3. FUNDAMENTOS

#### DICCIONARIO:

Un diccionario es una colección de elementos que se relacionan con una clave única. Cada elemento en un diccionario consta de una clave y un valor asociado a esa clave. Las claves son únicas e inmutables, mientras que los valores pueden ser de cualquier tipo de datos mutable o inmutable. En Python, se define un diccionario con llaves { } y se separan las claves y los valores con ":". Por ejemplo: `mi_diccionario = {"clave1": valor1, "clave2": valor2, "clave3": valor3}`

#### LISTAS:

Por otro lado, una lista es una colección de elementos ordenados y mutables. Cada elemento en una lista se identifica por su posición o índice. En Python, se define una lista con corchetes [ ] y se separan los elementos con comas. Por ejemplo:

`mi_lista = [elemento1, elemento2, elemento3]`

### 4. PROCEDIMIENTO

#### MENU

1.- Agregar (automático)

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 fábrica.

NOTA 2: VALIDAR EL PROGRAMA 100%

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



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 5. RESULTADOS Y CONCLUSIONES



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

Este es uno de los ejercicios con los que mas eh batallado pero aun así se logró solucionar esta actividad sin problemas y 100% validado.

```
import yaml
import random as rm
import json
import pandas as pd
diccionario = {}
id = 0
lista = []
registro_id = set()

[2] n_fem = ["Ana", "Beatriz", "Carla", "Diana", "Elena", "Laura", "Martina", "Natalia", "Victoria Elena", "María Paula", "Carolina", "María Fernanda"]
n_mas = ["Juan Pablo", "Luis Miguel", "Jorge Luis", "José Luis", "Adrián", "Alejandro", "Carlos", "César", "David", "Emilio", "Federico", "Gabriel"]
puesto = ["operador", "supervisor", "jefe de produccion", "jefe de mantenimiento", "jefe de calidad", "gerente", "tecnico", "jefe de logistica"]
apellido = ["Perez", "Galindo", "Nevarez", "Yepiz", "Zambrano", "Cardenas", "Garza", "Morro", "Murrieta", "Vazquez", "Martinez", "Lopez", "Palma",
```

```
def agg_auto():

    global id, diccionario, lista
    id += 1

    if 'lista' not in globals():
        lista = []
    if not lista:
        id = 0

    while True:
        nuevo_id = rm.randint(1, 10000) # Generar un id aleatorio

        if nuevo_id not in registro_id: # Verificar si el id generado ya está en el registro
            registro_id.add(nuevo_id) # Agregar el nuevo id al registro
            break

    diccionario = {"id": nuevo_id, "Nombre": "", "Apellido Paterno": "", "Apellido Materno": "", "Edad": 0, "Puesto": "", "Salario": 0}
    diccionario['id'] = nuevo_id

    # Nombres
    nombre_aleatorio = rm.choice(n_fem + n_mas)
    diccionario['Nombre'] = nombre_aleatorio

    # Apellidos
    ap_paterno = rm.choice(apellido)
    ap_materno = rm.choice(apellido)
    while ap_paterno == ap_materno:
        ap_materno = rm.choice(apellido)

    diccionario['Apellido Paterno'] = ap_paterno
    diccionario['Apellido Materno'] = ap_materno

    # Edad
    edad = rm.randint(18, 61)
    diccionario['Edad'] = edad
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
# Puestos
puest = rm.choice(puesto)
diccionario['Puesto'] = puest

# Salarios
salario = rm.randint(10000, 50000)
diccionario['Salario'] = salario

print("Diccionario creado \n")

return diccionario
```

```
[4] def validar_numero(mensaje):
    while True:
        try:
            valor = int(input(mensaje))
            return valor
        except ValueError:
            print("Error: Solo debes ingresar valores numericos\n")
```

```
def agg_manual():
    global id, diccionario, lista

    if 'lista' not in globals():
        lista = []

    diccionario = {}

    while True:
        idd = validar_numero("Ingrese el id: ")

        if idd not in registro_id: # Verificar si el id ya está en el registro
            registro_id.add(idd) # Agregar el id al registro
            break

        print("El id ingresado ya existe en el registro, por favor ingrese uno nuevo.")
        nombre = input("Ingrese el nombre: ")
        appaterno = input("Ingrese el apellido paterno: ")
        apmaterno = input("Ingrese el apellido materno: ")
        edadd = validar_numero("Ingrese la edad: ")
        puest = input("Ingrese el puesto: ")
        sala = validar_numero("Ingrese el salario: ")

        claves = ["id", "Nombre", "Apellido Paterno", "Apellido Materno", "Edad", "Puesto", "Salario"]
        valores = [idd, nombre, appaterno, apmaterno, edadd, puest, sala]
        diccionario = dict(zip(claves, valores))
        print("Diccionario creado \n")
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

✓  
0 s



```
return diccionario
```

✓  
0 s

```
[6] def imprimir_lista(lista=None):  
    if lista is None:  
        print("La lista no existe.")  
        return  
    elif not lista:  
        print("La lista está vacía.")  
        return  
    a = json.dumps(lista, sort_keys=False, indent = 4)  
    b = yaml.dump(lista, sort_keys=False, default_flow_style=False)  
    c = pd.DataFrame(lista)  
    impresion = [a, b, c]  
    aleatorio = rm.choice(impresion)  
    return print(aleatorio)
```

✓  
0 s

```
[7] def buscar_id(list, id_buscado):  
    for elemento in list:  
        if elemento['id'] == id_buscado:  
            return elemento  
    return None
```

✓  
0 s

```
[8] def eliminar_id(lista, id_eliminado):  
    if lista is None:  
        print("La lista no existe.")  
        return False  
  
    for i in range(len(lista)):  
        if lista[i]["id"] == id_eliminado:  
            del lista[i]  
            print(f"El elemento con ID {id_eliminado} ha sido eliminado.")  
            return True  
  
    print(f"No se encontró ningún elemento con el ID {id_eliminado}.")  
    return False
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
✓ [9] def ordenar_lista(lista=None):  
0s  
    if lista is None:  
        print("La lista no existe.")  
        return  
    elif not lista:  
        print("La lista está vacía.")  
        return  
    lista_ordenada = sorted(lista, key=lambda x: x['id'])  
  
    if lista == lista_ordenada:  
        print("La lista ya esta ordenada")  
        return  
    lista = lista_ordenada  
    print("La lista a sido ordenada correctamente")  
    return lista
```

```
✓ [10]  
0s  
def menu():  
    print("MENU \n1.- Agregar automatico \n2.- Agregar Manual \n3.- Imprimir lista \n4.- B")  
    while True:  
        global lista, diccionario  
        try:  
            opcion = int(input("Ingrese una opcion: "))  
            if opcion == 1:  
                dicc = agg_auto()  
                lista.append(dicc)  
                continue  
            elif opcion == 2:  
                dicc = agg_manual()  
                lista.append(dicc)  
                continue  
            elif opcion == 3:  
                imprimir_lista(lista)  
                continue  
            elif opcion == 4:  
                id_buscado = int(input("Ingrese el ID a buscar: "))  
                elemento_encontrado = buscar_id(lista, id_buscado)  
                if elemento_encontrado:
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
if elemento_encontrado:
    print(elemento_encontrado)
else:
    print("El ID no fue encontrado en la lista")
    continue
elif opcion == 5:
    lista = ordenar_lista(lista)
    continue
elif opcion == 6:
    id_eliminar = validar_numero("Ingresa el ID a eliminar")
    try:
        eliminar_id(lista, id_eliminar)
    except UnboundLocalError:
        print("Error: El ID no existe en la lista.")
        continue
elif opcion == 7:
    del lista
    print("Lista eliminada con exito \n")
    continue
elif opcion == 0:
    print("Saliendo del MENU...")
    break
else:
    print("Opción inválida. Por favor, ingrese una opción válida.")
except ValueError:
    print("Error: Solo debes de ingresar valores numericos")
except NameError:
    print("Error: La lista no existe")
```

### 6. ANEXOS

[https://github.com/a375295/ProgramacionPYTHON/blob/main/NPE\\_PY\\_A10.ipynb](https://github.com/a375295/ProgramacionPYTHON/blob/main/NPE_PY_A10.ipynb)

### 7. REFERENCIAS

#### DICCIONARIOS Y LISTAS EN PYTHON

dice, Z. et al. (2022) Diccionarios y Listas en python, Tutorial python. Available at:

<https://tutorialpython.com/listas-en-python/> (Accessed: April 16, 2023).