



# 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. 7**

**Tema - Unidad:** Ciclos y Funciones

**Ensenada Baja California a 21 de marzo del 2023**



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 1. INTRODUCCION

Los temas que serán vistos serán las funciones random, range y listas

### 2. COMPETENCIA

Aprenderemos a dominar tanto las funciones mencionadas como el ciclo común, realizando problemas y practicando.

### 3. FUNDAMENTOS

Las funciones **random** en Python se utilizan para generar números aleatorios. El módulo de Python "**random**" proporciona una variedad de funciones para trabajar con números aleatorios. Estas funciones se utilizan para generar números enteros aleatorios, números de punto flotante aleatorios y secuencias aleatorias.

La función **range** en Python se utiliza para generar una secuencia de números. La sintaxis de la función range es range (start, stop, step). La función range genera una secuencia de números que comienza en "start", termina en "stop" y se incrementa en "step". Por defecto, el valor de "start" es 0 y el valor de "step" es 1.

En Python, una **lista** es un tipo de estructura de datos que se utiliza para almacenar una colección de elementos en un orden determinado. Las listas son muy versátiles y se utilizan comúnmente en la programación de Python para manejar datos de manera eficiente.

**¿Qué son y uso de las listas?** Las listas en Python son colecciones ordenadas de elementos, que se pueden modificar. Las listas son útiles para almacenar colecciones de datos relacionados, como una lista de números o una lista de nombres. Las listas son similares a los arrays en otros lenguajes de programación, pero tienen algunas características adicionales que los hacen más útiles.



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 4. PROCEDIMIENTO

#### Problemas:

1.- Función que utilice una lista con los nombres de tus mascotas, o artistas favoritos, (minimo 5, maximo 10) imprimir las cadenas y la cantidad de caracteres de cada cadena.

EJEMPLO SALIDA:

[0]SCARLETT --> 8 CARACTERES

[1]GATINA --> 6 CARACTERES

[2]PELUSA --> 6 CARACTERES

[3]PELUSINA --> 8 CARACTERES

[4]MILO --> 4 CARACTERES

[5]KHALIFA --> 7 CARACTERES

2.- Programa que utilice 2 funciones.

Función que genere y regrese una lista con 10 números aleatorios entre el 30 y 50 (no repetidos).

Función que reciba una lista , Imprimir la lista (Indice y Valor )

Mandar llamar la función e imprimir toda la lista.

3.- Escribe una función que reciba dos listas de números del mismo tamaño y calcule la suma de los elementos correspondientes de cada lista. (regresar una nueva lista) . Luego, muestra una lista con los resultados de cada suma. Nota: si las listas no son del mismo tamaño mandar msge y utilizar el tamaño de la lista mas pequeña)

4.- Escribe una función llamada eliminar\_duplicados que reciba una lista como parámetro y elimine los elementos duplicados. El resultado debe ser una nueva lista sin duplicados. Pide al usuario que ingrese una lista y luego llama a la función eliminar\_duplicados para mostrar la lista sin duplicados. (Solo Numeros enteros, validar)

5.- Escribe una función que calcule la media y la mediana de una lista de números enteros.

La media es la suma de todos los elementos y dividido entre la cantidad de elementos

La mediana es el valor que queda en la mitad de la lista cuando se ordena de forma

ascendente. Si la lista tiene un número par de elementos, la mediana se calcula como el promedio de los dos valores centrales. Puedes utilizar el método sort() para ordenar la lista y luego calcular la mediana según el tamaño de la lista.



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

6.- Escribe una función que calcule la media móvil de una lista de números enteros. La media móvil es el promedio de un número determinado de elementos consecutivos de la lista. Puedes utilizar un ciclo y una variable de suma para calcular la media móvil de cada sublista y guardar los resultados en otra lista.

### 5. RESULTADOS Y CONCLUSIONES

A continuación, mostrare las funciones que se desarrollaron para cada uno de los problemas.

```
# ACTIVIDAD 7: LISTAS

[17] import random

#1.- Función que utilice una lista con los nombres de tus mascotas, o artistas favoritos,
#(mínimo 5, máximo 10) imprimir las cadenas y la cantidad de caracteres de cada cadena.

def imprimir_carac(lista):
    try:
        contador = 0
        for elemento in lista:
            caracteres = len(elemento)
            print(f"[{contador}] {elemento} --> {caracteres} caracteres.")
            contador += 1
        except Exception as e:
            print(f"Ocurrió un error: {e}")

    lista = ["SCARLETT", "GATINA", "PELUSA", "PELUSINA", "MILO", "KHALIFA"]
    imprimir_carac(lista)

#2.- Programa que utilice 2 funciones,
#Función que genere y regrese una lista con 10 números aleatorios entre el 30 y 50 (no repetidos).
#Función que reciba una lista, Imprimir la lista (Índice y Valor)
#Mandar llamar la función e imprimir toda la lista.

def generar_lista_aleatoria():
    try:
        lista = random.sample(range(30, 51), 10)
        return lista
    except Exception as e:
        print(f"Ocurrió un error: {e}")

def imprimir_lista(lista):
    try:
        for i, valor in enumerate(lista):
            print(f"{i}. {valor}")
    except Exception as e:
        print(f"Ocurrió un error: {e}")

lista_aleatoria = generar_lista_aleatoria()
imprimir_lista(lista_aleatoria)
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

#3.- Escribe una función que reciba dos listas de números del mismo tamaño y calcule la suma de los elementos correspondientes de cada lista. (regresar una nueva lista) . Luego, muestra una lista con los resultados de cada suma.  
#Nota: si las listas no son del mismo tamaño mandar msge y utilizar el tamaño de la lista mas pequeña)

```
def sumar_listas(lista1, lista2):
    try:
        if len(lista1) != len(lista2):
            print("Las listas no son del mismo tamaño. Se usará el tamaño de la lista más pequeña.")
            tamaño_minimo = min(len(lista1), len(lista2))
            lista1 = lista1[:tamaño_minimo]
            lista2 = lista2[:tamaño_minimo]
            resultado = []
            for i in range(len(lista1)):
                suma = lista1[i] + lista2[i]
                resultado.append(suma)
            return resultado
        except Exception as e:
            print(f"Ocurrió un error: {e}. Solo se aceptan valores numericos")

lista1 = [1, 2, 3, 4, 5]
lista2 = [10, 20, 30, 40]

resultado = sumar_listas(lista1, lista2)
if resultado:
    print(resultado)
```

# 4.- Escribe una función llamada eliminar\_duplicados que reciba una lista como parámetro y elimine los elementos duplicados. El resultado debe ser una nueva lista sin duplicados. #Pide al usuario que ingrese una lista y luego llama a la función eliminar\_duplicados para mostrar la lista sin duplicados. # (Solo Números enteros, validar)

```
def eliminar_duplicados(lista):
    try:
        # Validar que todos los elementos sean números enteros
        for elem in lista:
            if not isinstance(elem, int):
                raise ValueError("La lista solo debe contener números enteros.")

        # Eliminar los elementos duplicados utilizando un conjunto
        lista_sin_duplicados = list(set(lista))
        return lista_sin_duplicados
    except ValueError as ve:
        print(f"Ocurrió un error: {ve}")
    except Exception as e:
        print(f"Ocurrió un error: {e}")

lista = []
while True:
    try:
        entrada = input("Ingrese un número entero para la lista (Enter para terminar): ")
        if entrada == "":
            break
        elemento = int(entrada)
        lista.append(elemento)
    except ValueError:
        print("Por favor, ingrese solo números enteros.")

lista_sin_duplicados = eliminar_duplicados(lista)
if lista_sin_duplicados:
    print(f"La lista sin duplicados es: {lista_sin_duplicados}")
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
# 5.- Escribe una función que calcule la media y la mediana de una lista de números enteros.
#La media es la suma de todos los elementos y dividido entre la cantidad de elementos
#La mediana es el valor que queda en la mitad de la lista cuando se ordena de forma ascendente.
#Si la lista tiene un número par de elementos, la mediana se calcula como el promedio de los dos valores centrales.
#Puedes utilizar el método sort() para ordenar la lista y luego calcular la mediana según el tamaño de la lista.
def calcular_media_mediana(lista):
    try:
        lista.sort() # ordenar la lista de menor a mayor
        n = len(lista)
        # Calcular la media
        media = sum(lista) / n

        # Calcular la mediana
        if n % 2 == 0:
            mediana = (lista[n//2] + lista[n//2 - 1]) / 2
        else:
            mediana = lista[n//2]

        return media, mediana

    except TypeError:
        print("La lista debe contener sólo números enteros")
    except ZeroDivisionError:
        print("La lista no puede estar vacía")

lista = [3, 5, 1, 8, 2, 9, 6]
calcular_media_mediana(lista)
```

```
# 6.- Escribe una función que calcule la media móvil de una lista de números enteros.
# La media móvil es el promedio de un número determinado de elementos consecutivos de la lista.
# Puedes utilizar un ciclo y una variable de suma para calcular la media móvil de cada sublista y guardar los resultados en otra lista.
def calcular_media_movil(lista, n):
    try:
        media_movil = []
        for i in range(len(lista)-n+1):
            sublista = lista[i:i+n]
            suma = sum(sublista)
            promedio = suma / n
            media_movil.append(promedio)
        return media_movil

    except TypeError:
        print("La lista debe contener sólo números enteros")
    except ZeroDivisionError:
        print("La lista no puede estar vacía")

lista = [3, 5, 1, 8, 2, 9, 6]
calcular_media_movil(lista, 3)
```

## 6. ANEXOS

[https://drive.google.com/file/d/1rltOFNVxJ4IE3alHFDz2Btn0ZeknWp3y/view?usp=share\\_link](https://drive.google.com/file/d/1rltOFNVxJ4IE3alHFDz2Btn0ZeknWp3y/view?usp=share_link)

## 7. REFERENCIAS

### Listas en Python

[https://drive.google.com/file/d/1x9qYezVoNTCuD4vG6EHI9SMDdbUQHDwo/view?usp=share\\_link](https://drive.google.com/file/d/1x9qYezVoNTCuD4vG6EHI9SMDdbUQHDwo/view?usp=share_link)