**\*\*Listas:\*\***

- `append(x)`: Añade un elemento al final de la lista.

- `extend(iterable)`: Extiende la lista agregando todos los elementos del iterable.

- `insert(i, x)`: Inserta un elemento en la posición dada.

- `remove(x)`: Elimina el primer elemento cuyo valor sea x.

- `pop([i])`: Elimina el elemento en la posición dada y lo devuelve.

- `clear()`: Elimina todos los elementos de la lista.

- `index(x[, start[, end]])`: Devuelve el índice del primer elemento cuyo valor sea x.

- `count(x)`: Devuelve el número de veces que x aparece en la lista.

- `sort(key=None, reverse=False)`: Ordena los elementos de la lista.

- `reverse()`: Invierte los elementos de la lista.

- `copy()`: Devuelve una copia superficial de la lista.

**\*\*Sets:\*\***

- `add(elem)`: Añade un elemento al set.

- `remove(elem)`: Elimina un elemento del set. Lanza un error si no existe.

- `discard(elem)`: Elimina un elemento del set si está presente.

- `pop()`: Elimina y devuelve un elemento aleatorio del set.

- `clear()`: Elimina todos los elementos del set.

- `union(\*others)`: Devuelve un nuevo set con elementos de ambos sets.

- `intersection(\*others)`: Devuelve un nuevo set con elementos comunes entre los sets.

- `difference(\*others)`: Devuelve un nuevo set con elementos que no están en los otros.

- `symmetric\_difference(other)`: Devuelve un nuevo set con elementos que están en uno de los sets pero no en ambos.

- `update(\*others)`: Actualiza el set añadiendo elementos de otros sets.

**\*\*Diccionarios:\*\***

- `clear()`: Elimina todos los elementos del diccionario.

- `copy()`: Devuelve una copia superficial del diccionario.

- `fromkeys(seq[, value])`: Crea un nuevo diccionario con claves de seq y valores iguales a value.

- `get(key[, default])`: Devuelve el valor para key si key está en el diccionario, sino default.

- `items()`: Devuelve una vista de los pares clave-valor del diccionario.

- `keys()`: Devuelve una vista de las claves del diccionario.

- `pop(key[, default])`: Elimina la clave dada y devuelve su valor.

- `popitem()`: Elimina y devuelve un par (clave, valor) aleatorio.

- `setdefault(key[, default])`: Devuelve el valor de la clave si está en el diccionario, sino inserta la clave con un valor default.

- `update([other])`: Actualiza el diccionario con los pares clave-valor de other.

- `values()`: Devuelve una vista de todos los valores del diccionario.

**\*\*Strings:\*\***

- `capitalize()`: Convierte el primer carácter a mayúscula.

- `casefold()`: Convierte string a minúsculas para comparaciones sin distinción de mayúsculas y minúsculas.

- `center(width[, fillchar])`: Centra el string, rellenando con fillchar.

- `count(sub[, start[, end]])`: Cuenta las ocurrencias de sub en el string.

- `encode(encoding=”utf-8”, errors=”strict”)`: Codifica el string usando el codificador especificado.

- `endswith(suffix[, start[, end]])`: Verifica si el string termina con el sufijo especificado.

- `expandtabs(tabsize=8)`: Reemplaza los tabs en el string por espacios.

- `find(sub[, start[, end]])`: Busca sub en el string y devuelve el índice más bajo donde se encuentra.

- `format(\*args, \*\*kwargs)`: Formatea el string usando la formatación de campos de reemplazo.

- `join(iterable)`: Une los elementos de iterable en un único string, separados por el string sobre el que se llama el método.

- `lower()`: Convierte todos los caracteres en minúsculas.

- `upper()`: Convierte todos los caracteres en mayúsculas.

- `replace(old, new[, count])`: Reemplaza todas las ocurrencias de old por new en el string.

- `split(sep=None, maxsplit=-1)`: Divide el string en una lista de strings usando sep como delimitador.

- `strip([chars])`: Elimina los caracteres especificados del inicio y final del string.

**\*\*Funciones Incorporadas que se Comportan como Métodos Estáticos:\*\***

- `abs()`: Devuelve el valor absoluto de un número.

- `all()`: Retorna `True` si todos los elementos de un iterable son verdaderos.

- `any()`: Retorna `True` si algún elemento de un iterable es verdadero.

- `ascii()`: Retorna una representación de un objeto que es una cadena ASCII.

- `bin()`: Convierte un número entero en una cadena binaria.

- `bool()`: Convierte un valor en un booleano.

- `bytearray()`: Crea y retorna un nuevo arreglo de bytes.

- `bytes()`: Crea y retorna un nuevo objeto de bytes.

- `callable()`: Verifica si el objeto parece llamable.

- `chr()`: Retorna el string que representa un carácter cuyo Unicode es el número entero dado.

- `classmethod()`: Transforma un método en un método de clase.

- `compile()`: Compila el código fuente en un objeto de código o AST.

- `complex()`: Crea un número complejo.

- `delattr()`: Elimina un atributo de un objeto.

- `dict()`: Crea un diccionario.

- `dir()`: Intenta retornar una lista de atributos válidos para ese objeto.

- `divmod()`: Toma dos números y devuelve un par de números que consiste en su cociente y resto.

- `enumerate()`: Retorna un objeto enumerado.

- `eval()`: Ejecuta código Python dentro de un programa.

- `exec()`: Ejecuta código Python dinámicamente.

- `filter()`: Construye un iterador a partir de los elementos de iterable para los que la función retorna verdadero.

- `float()`: Convierte una cadena o número en un número de punto flotante.

- `format()`: Formatea un valor en una cadena.

- `frozenset()`: Crea un nuevo conjunto inmutable.

- `getattr()`: Retorna el valor del atributo nombrado de un objeto.

- `globals()`: Retorna un diccionario que representa la tabla de símbolos global actual.

- `hasattr()`: Retorna `True` si el objeto tiene el atributo nombrado.

- `hash()`: Retorna el valor hash de un objeto.

- `help()`: Invoca el sistema de ayuda integrado.

- `hex()`: Convierte un número entero en una cadena hexadecimal.

- `id()`: Retorna la “identidad” de un objeto.

- `input()`: Permite al usuario ingresar datos.

- `int()`: Convierte una cadena o número en un número entero.

- `isinstance()`: Verifica si un objeto es una instancia de una clase o de una subclase de ella.

- `issubclass()`: Verifica si una clase es una subclase de otra.

- `iter()`: Retorna un objeto iterador.

- `len()`: Retorna la longitud (el número de elementos) de un objeto.

- `list()`: Crea una nueva lista.

- `locals()`: Actualiza y retorna un diccionario que representa la tabla de símbolos local actual.

- `map()`: Aplica una función a todos los elementos de un iterable.

- `max()`: Retorna el mayor elemento de un iterable o el mayor de dos o más argumentos.

- `min()`: Retorna el menor elemento de un iterable o el menor de dos o más argumentos.

- `next()`: Recupera el siguiente elemento de un iterador.

- `object()`: Crea un nuevo objeto sin características.

- `oct()`: Convierte un número entero en una cadena octal.

- `open()`: Abre un archivo y retorna un objeto de archivo correspondiente.

- `ord()`: Retorna un entero que representa el punto de código Unicode del carácter dado.

- `pow()`: Retorna la potencia de un número.

- `print()`: Imprime objetos a la salida de texto.

- `property()`: Retorna un objeto de propiedad.

- `range()`: Retorna un objeto de rango inmutable.

- `repr()`: Retorna una cadena que representa un objeto.

- `reversed()`: Retorna un iterador inverso.

- `round()`: Redondea un número a un número específico de dígitos.

- `set()`: Crea un nuevo conjunto.

- `setattr()`: Establece el valor de un atributo de un objeto.

- `slice()`: Retorna un objeto de rebanada.

- `sorted()`: Retorna una nueva lista ordenada a partir de los elementos de un iterable.

- `staticmethod()`: Transforma un método en un método estático.

- `str()`: Crea una nueva cadena de caracteres.

- `sum()`: Suma los elementos de un iterable.

- `super()`: Retorna un objeto de proxy que delega llamadas de método a una clase padre o hermano.

- `tuple()`: Crea una nueva tupla.

- `type()`: Retorna el tipo de un objeto o crea un nuevo tipo.

- `vars()`: Retorna el diccionario de atributos de un módulo, clase, instancia o cualquier otro objeto con un diccionario de atributos.

- `zip()`: Retorna un iterador de tuplas, donde la i-ésima tupla contiene el i-ésimo elemento de cada uno de los argumentos iterables.