

**pop(list.pop(i)):**El método elimina el elemento en el índice dado de la lista y devuelve el elemento eliminado.

**append(list.append(elemento)):**Agrega un objeto (un número, una cadena o un otra lista) al final de mi\_lista

**remove(list.remove(elemento)):**Elimina el primer elemento coincidente (que se pasa como argumento) de la lista.

**insert(list.insertar(i, elem)):**El método inserta un elemento en la lista en el índice especificado.

**reverse(list.reverse()):**El método invierte los elementos de la lista.

**extend(list1.extend(i)):**agrega todos los elementos de un iterable (lista, tupla, cadena, etc.) al final de la lista.

**copy(new\_list = list.copy()):**método devuelve una copia superficial de la lista.

**clear(list.borrar()):**El método elimina todos los elementos de la lista.

**max:**La función devuelve el elemento más grande en un iterable.

**reversed(reversed(sequence\_object)):** El método calcula el reverso de un objeto de secuencia dado y lo devuelve en forma de lista.

**map(mapa (función, iterable, ...)):** function aplica una función determinada a cada elemento de un iterable (lista, tupla, etc.) y devuelve un iterador.

**min:**La función devuelve el elemento más pequeño en un iterable.

**slice(slice(start, stop, step)):** La función devuelve un objeto de división que se usa para dividir cualquier secuencia (cadena, tupla, lista, rango o bytes).

**sorted(sorted(iterable, key=None, reverse=False)):** La función ordena los elementos de un iterable dado en un orden específico (ascendente o descendente) y lo devuelve como una lista.

**continue:** sigue con la próxima instrucción

**break:** sale del loop

**sort(sorted(list, key=..., reverse=...)):**El método ordena los elementos de una lista en orden ascendente o descendente.

**filter(filter(function, iterable)):** La función extrae elementos de un iterable (lista, tupla, etc.) para los cuales una función devuelve True.

**lambda:** Función anónima

**index(list.index(element, start, end)):** Método devuelve el índice del elemento especificado en la lista.

**isdecimal(string.isdecimal()):** El método isdecimal() devuelve True si todos los caracteres de una cadena son caracteres decimales. Si no, devuelve False.

**capitalize(string.capitalize()):** El método convierte el primer carácter de una cadena en una letra mayúscula y todos los demás alfabetos en minúsculas.

**isalpha:** El método isalpha() devuelve True si todos los caracteres de la cadena son alfabetos. Si no, devuelve False.

**isdigit(cadena.isdigit()):** El método devuelve True si todos los caracteres de una cadena son dígitos. Si no, devuelve False.

**zip:** La función toma iterables (pueden ser cero o más), los agrega en una tupla y lo devuelve.

**pass:** usamos el `pass` para construir un cuerpo que no hace nada.

**`upper(string.upper())`:** El método convierte todos los caracteres en minúsculas de una cadena en caracteres en mayúsculas y los devuelve.

**`lower(string.lower())`:** El método convierte todos los caracteres en mayúsculas de una cadena en caracteres en minúsculas y los devuelve.

**`len`:** La función devuelve el número de elementos (longitud) en un objeto.

**`sum(sum(iterable, start))`:** La función agrega los elementos de un iterable y devuelve la suma.

**`split(str.split(separator, maxsplit))`:** El método divide una cadena en el separador especificado y devuelve una lista de cadenas.

**`join(string.join(iterable))`:** El método devuelve una cadena al unir todos los elementos de un iterable (lista, cadena, tupla), separados por el separador dado.

**`all`:** función devuelve `True` si todos los elementos en el iterable dado son verdaderos. Si no, devuelve `False`.

**`any`:** función devuelve `True` si cualquier elemento de un iterable es `True`. Si no, devuelve `False`.