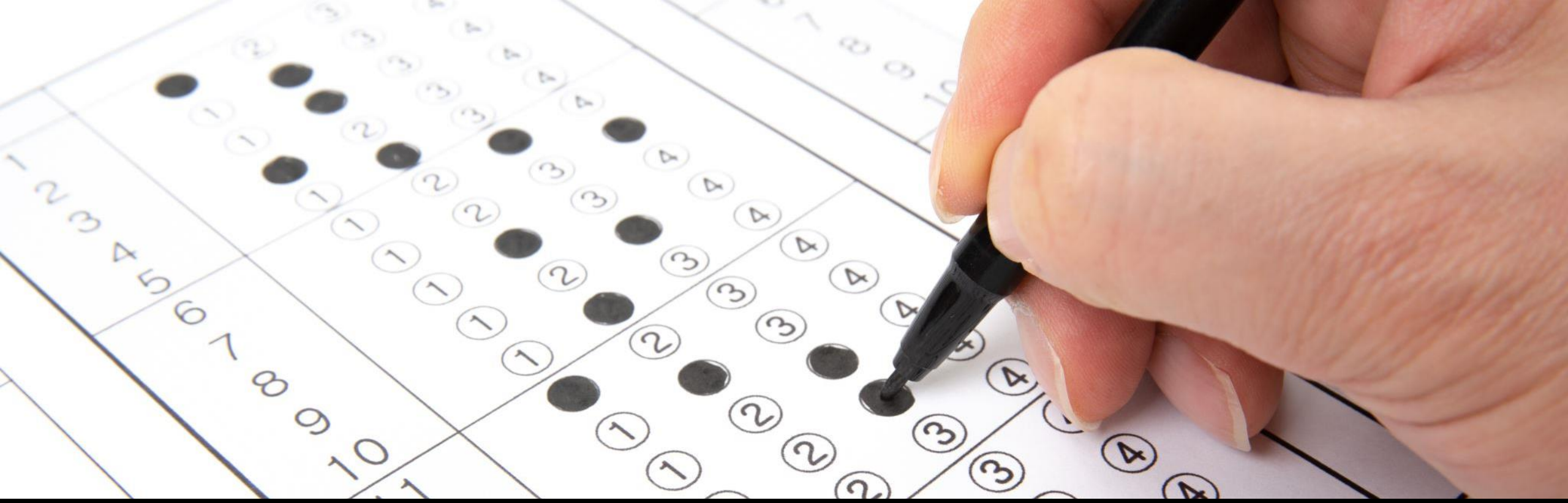

LIST

Built-in Functions & list methods

PYTHON'S BUILT-IN FUNCTIONS ON LIST

- `len(L)` gives the number of elements in `L`
- `sorted(L)` takes a list `L` and returns a new sorted list, leaving the original list `L` unchanged.
- `min(L)` returns the smallest element in the list.
- `max(L)` returns the largest element in the list.
- `sum(L)` returns the sum of all elements in the list.

Note: The elements in the list should be *compatible* with comparison operations.

Also `sum(L)` requires the elements to be *numbers*.

LIST IS A CLASS

- List also a **class**. Individual lists are **objects** belonging to the list class.
- The list class defines numerous methods that allow you to perform operations on lists, such as adding or removing elements, sorting, searching, and more.
- For example, `L.sort()` is a method that sorts the list `L` in-place, meaning it modifies the original list `L`.
- This method does not return a new list but rather sorts the existing list `L` directly.

LIST METHODS

Method	Descriptions
`append(elem)`	adds an element to the end of the list
`extend(list)`	adds all elements of a list to another list
`insert(i, elem)`	inserts `elem` at the defined index, elements after elem are shifted right
`remove(elem)`	removes `elem` from the list
`pop(i)`	**returns** and removes an element at the given index i
`clear()`	removes all items from the list
`index(elem)`	**returns** the **index** of the *first* matched item
`count(elem)`	**returns** the number of `item`'s in the list
`sort()`	sort items in a list in ascending order
`reverse()`	reverse the order of items in the list
`copy()`	**returns** a copy of the list

LISTS ARE MUTABLE

- Unlike strings, lists are ***mutable***, meaning they can be changed after creation.

```
>>> L = [1,2,3]
>>> L[1] = 5
>>> L
[1, 5, 3]
```

EXAMPLE: INPUT + LIST

- Lets see another example on how to combine inputs + lists
- See File: L6E4.py