10. Create four modules as list1.py, set2.py, dict3.py, m4.py

a. list1.py contains append1(x), extend1(l), pop(), remove1(x)

# list1.py

# Internal list to work with

\_internal\_list = []

def append1(x):

"""Append an element to the internal list."""

\_internal\_list.append(x)

def extend1(l):

"""Extend the internal list with another list."""

if isinstance(l, list):

\_internal\_list.extend(l)

else:

raise TypeError("extend1() argument must be a list")

def pop():

"""Pop the last element from the internal list."""

if \_internal\_list:

return \_internal\_list.pop()

else:

raise IndexError("pop from empty list")

def remove1(x):

"""Remove first occurrence of x from the internal list."""

if x in \_internal\_list:

\_internal\_list.remove(x)

else:

raise ValueError(f"{x} not found in list")

def get\_list():

"""Return a copy of the current list."""

return \_internal\_list.copy()