**List ADT using Array in Python with insert, delete, search and modify operations**

class ArrayList:

def \_\_init\_\_(self, max\_size):

self.max\_size = max\_size

self.array = [None] \* max\_size

self.current\_size = 0

def is\_empty(self):

return self.current\_size == 0

def is\_full(self):

return self.current\_size == self.max\_size

def insert\_end(self, value):

if self.is\_full():

return "List is full. Can't Insert."

self.array[self.current\_size] = value

self.current\_size += 1

def delete\_end(self):

if self.is\_empty():

return "List is Empty. Can't Delete."

self.array[self.current\_size - 1] = None

self.current\_size -= 1

def insert\_front(self, value):

if self.is\_full():

return "List is full. Can't Insert."

for i in range(self.current\_size, 0, -1):

self.array[i] = self.array[i - 1]

self.array[0] = value

self.current\_size += 1

def delete\_front(self):

if self.is\_empty():

return "List is Empty. Can't Delete."

for i in range(1, self.current\_size):

self.array[i - 1] = self.array[i]

self.array[self.current\_size - 1] = None

self.current\_size -= 1

def insert\_at(self, index, value):

if self.is\_full():

return "List is full. Can't Insert."

if index > self.current\_size or index < 0:

return "nth Position Not available in List"

for i in range(self.current\_size, index, -1):

self.array[i] = self.array[i - 1]

self.array[index] = value

self.current\_size += 1

def delete\_at(self, index):

if self.is\_empty():

return "List is Empty. Can't Delete."

if index >= self.current\_size or index < 0:

return "nth Position Not available in List"

for i in range(index, self.current\_size - 1):

self.array[i] = self.array[i + 1]

self.array[self.current\_size - 1] = None

self.current\_size -= 1

def search(self, value):

if self.is\_empty():

return "List is Empty."

for i in range(self.current\_size):

if self.array[i] == value:

return f"Search Element available in Position {i + 1}"

return "Search Element not available in the List"

def print\_list(self):

if self.is\_empty():

return "List is Empty."

elements = [self.array[i] for i in range(self.current\_size)]

return elements

# Example usage

if \_\_name\_\_ == "\_\_main\_\_":

max\_size = 10

my\_list = ArrayList(max\_size)

# Test insert end

my\_list.insert\_end(10)

my\_list.insert\_end(20)

my\_list.insert\_end(30)

print("After insertions at end:", my\_list.print\_list())

# Test delete end

my\_list.delete\_end()

print("After deletion from end:", my\_list.print\_list())

# Test insert front

my\_list.insert\_front(5)

print("After insertion at front:", my\_list.print\_list())

# Test delete front

my\_list.delete\_front()

print("After deletion from front:", my\_list.print\_list())

# Test insert at position

my\_list.insert\_at(1, 15)

print("After insertion at position 2:", my\_list.print\_list())

# Test delete at position

my\_list.delete\_at(1)

print("After deletion at position 2:", my\_list.print\_list())

# Test search

print(my\_list.search(20))

# Test print list

print("Elements in the list:", my\_list.print\_list())

**OUTPUT**

After insertions at end: [10, 20, 30]

After deletion from end: [10, 20]

After insertion at front: [5, 10, 20]

After deletion from front: [10, 20]

After insertion at position 2: [10, 15, 20]

After deletion at position 2: [10, 20]

Search Element available in Position 2

Elements in the list: [10, 20]