1. **Perform the CURD operation in Array**

class ArrCRUD:

def \_\_init\_\_(self, size):

self.arr = [None] \* size

self.count = 0

def insert(self, value): **🡪 ( inserting a new element )**

if self.count < len(self.arr):

self.arr[self.count] = value

self.count += 1

print(f"Inserted {value} at index {self.count - 1}")

else:

print("Array is full!")

def display(self): **🡪 (Displaying )**

print("Current Array:", [self.arr[i] for i in range(self.count)])

def update(self, index, value): **🡪 ( Updating )**

if 0 <= index < self.count:

old\_value = self.arr[index]

self.arr[index] = value

print(f"Updated index {index}: {old\_value} → {value}")

else:

print("Invalid index!")

def delete(self, index):  **🡪 ( Deleting )**

if 0 <= index < self.count:

deleted\_value = self.arr[index]

for i in range(index, self.count - 1):

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

self.arr[self.count - 1] = None

self.count -= 1

print(f"Deleted {deleted\_value} from index {index}")

else:

print("Invalid index!")

arr = ArrCRUD(5)

arr.insert(10)

arr.insert(20)

arr.insert(30)

arr.display()

arr.update(1, 50)

arr.display()

arr.delete(2)

arr.display()

1. **Take user input of numbers to perform linear search in an array or list**

def lin\_search(arr, target):

for index in range(len(arr)):

if arr[index] == target:

return f"Element {target} found at index {index}"

return f"Element {target} not found in the list."

arr = []

n = int(input("Enter the number of elements: "))

for i in range(n):

value = int(input(f"Enter element {i+1}: "))

arr.append(value)

target = int(input("Enter the number to search: "))

result = lin\_search(arr, target)

print(result)