2. Differentiate between **arrays and linked lists** in terms of:

   Memory allocation

   Arrays: Use the contiguous memory allocation whereby the entire array's size must be known at compile-time and cannot be easily changed during runtime.
   Linked Lists: Use the dynamic memory allocation whereby each node is allocated memory which can lead to non-contiguous memory usage but offer flexibility in size.

   Performance

   Arrays: Provide constant time complexity to random access since elements are stored contiguously and can be accessed directly via index.
   Linked Lists: Provide available sequential access to access a node, you may need to traverse the list from the beginning, which often leads to find any element in a linked list takes slow in term of time complexity.

   Insertion and deletion operations

   Arrays: Insertion of an element in an array refers to the operation of adding an element to the array while deleting an element from an array refers to the operation of the removal of an element from an array.
   Linked Lists: Insertion adds a new node to the linked list while deletion removes an existing node from the linked list.