Q #1) When is the Linked List used in Java?

Answer: As it is faster than collections like ArrayList in modification operations, it should be used in applications that require frequent addition/deletion operations. For applications that have mostly read-only data, ArrayList or similar collections can be used.

Q #2) What is ListNode?

Answer: A ListNode is a basic class associated with a linked list in Java and represents information associated with a single element or a node. Each ListNode consists of data and a pointer or reference to the next element.

Q #3) Does the Linked List allow null values?

Answer: Yes, the linked list allows any number of null values.

Q #4) What are the Advantages of a Linked List?

Answer: Some of the advantages are:

Manipulation operations like addition, deletion are faster in it.

There is no need to pre-allocate memory for a linked list and thus it results in efficient memory utilization.

It provides faster access time and without additional overhead for memory, and can be expanded in constant time.

It is a dynamic data structure

Grows and shrinks at run time depending on values added or deleted.

Q #5) What is the Application of the Linked List?

Answer: It is used mostly in the following applications:

To implement ‘undo’ functionality in software like MS-Word, Photoshop, etc.

To implement data structures like stack and queue.

We can also implement graphs using a linked list.

For bucket hashing, each bucket can be implemented as a linked list.

Q #6) What are the Limitations of a Linked List?

Answer: Some of the limitations are:

With an additional pointer to hold the reference of the next element in each node, memory utilized is much more than arrays.

This is a strictly sequentially accessed data structure hence nodes of the linked list must always be read from the beginning.

It is difficult to traverse it backward especially the singly-linked lists.

Since the nodes are stored in non-contiguous locations, the time required for access can be high