Activity about Singly linked list

1. What is a singly linked list, and how does it differ from an array?
  A singly linked list is a collection of nodes where each node contains data and a pointer that directs to the next node in the sequence. The first node is called the head, and the last node points to null (or None in Python) to indicate the end. Unlike arrays, linked lists do not store elements in consecutive memory locations. Instead, they work like a chain where you follow each link until you reach the end.

2. When would you prefer a linked list over an array, and vice versa?
  A linked list is more suitable when frequent insertions or deletions are needed since you can add or remove nodes without shifting other elements. On the other hand, an array is better when you need fast access to elements using indexes, or when the number of elements is fixed and predictable.

3. How are linked lists used in real-world applications (e.g., browser history, undo functionality)?
  Linked lists are useful in applications where data changes often, especially for adding, removing, or reordering elements. For instance, in browser history, each page visited can be linked to the next, making it easier to move back and forth. Similarly, in media playlists, songs or videos can be added or rearranged without much effort.

4. Cite your reference/s
 Malik, D. S. (2010). *Data Structures Using C++ (Second Edition)*. Course Technology.
 Weiss, M. A. (2013). *Data Structures and Algorithm Analysis in C++ (Fourth Edition)*. Pearson Education.