

Conceptual Question's Answers

1. Abstract Data Types (**ADT**) are data types that are defined by its behavior from users point of view. Examples for abstract data types will be arrays, lists, trees, stacks and so on. These data types implementation is independent of the type of data stored in each unit of the data type.
2. In a **queue** data added first is taken out of the queue first, in order. It can be visualized as if data is being added from the back and is being used and taken if from the front. In a **stack** data added first will be taken out last. It can be visualized as if books are being piled on top of each other. The book on top which is added last will be taken out first.
3. **3** different types of Data structures could be the following :-
 - **Trees** - Trees are data structures with parent child relationships where each unit of data is branched of from another unit. There can be *binary trees* where each unit will have a left-branch and right-branch or trees with more than 2 branches. Binary trees are more common and easy to implement.
 - **Linked List** - Linked lists are data structures that are comprised of nodes and linked/pointing to each other using pointers. Linked lists can be *singly linked lists* or *double linked lists*. In singly linked lists the current node points to the next node only whereas in doubly linked lists, one node will point to the next and previous nodes. Linked lists also have special nodes called the head and tail nodes which marks the start and end. A linked list cannot be indexed, but iterated.
 - **Stacks** - Stacks are *LIFO* data structure where the data added last will be popped out first. Stacks are pretty easy to implement and even used in lower levels of computing. When using a stack **push** method is implemented to add data to the top of the stack and **pop** method is implemented to remove data from the top of the stack.