**B-Tree**

• A B-tree is a tree data structure that keeps data sorted and allows searches, insertions, and deletions in logarithmic amortized time.

• It is most commonly used in database and file systems

• A self balanced search tree with multiple keys in every node and more than two children for every node.

• A B-tree of order M is a M-way search tree with following properties:

• Root can have 1 to M-1 keys.

• All nodes (except root node) have between (M-1)/2 and M-1 keys.

• All leaves are at same levels.

• If node have k children, then it has k-1 keys.

• Keys are stored in sorted order.

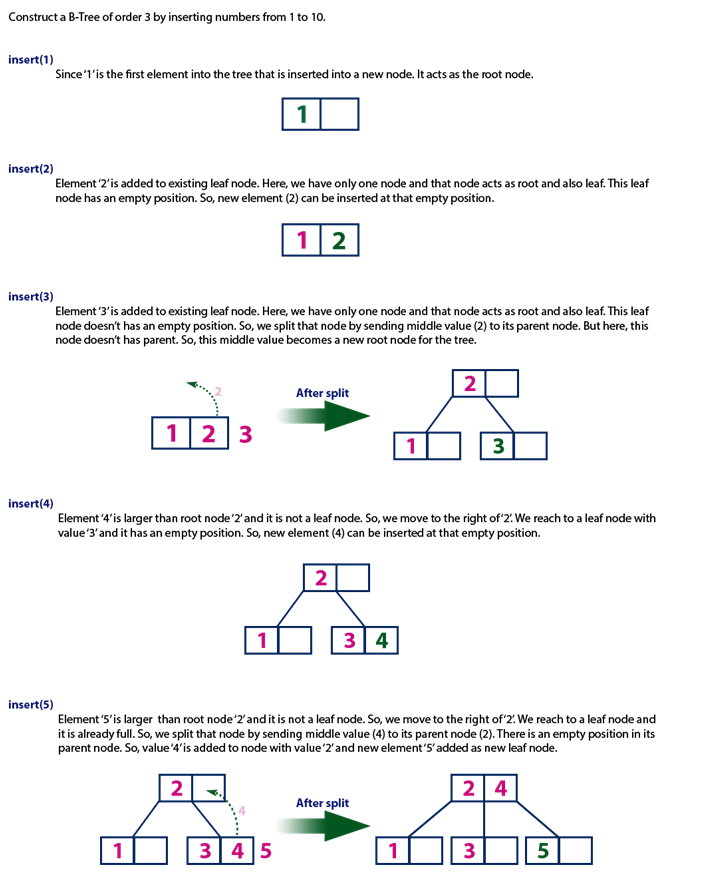
**Insertion Operation in B-Tree**

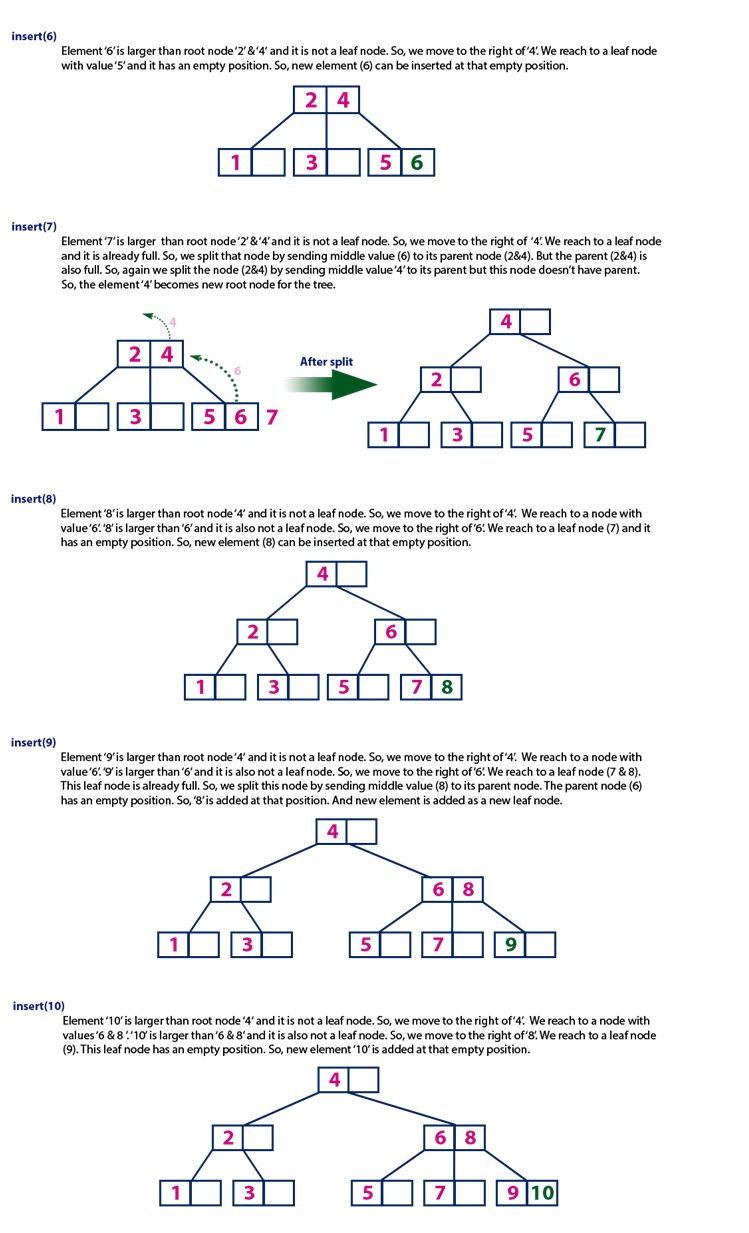
In a B-Tree, a new element must be added only at the leaf node. That means, the new key Value is always attached to the leaf node only. The insertion operation is performed as follows...

* **Step 1 -**Check whether tree is Empty.
* **Step 2 -**If tree is **Empty**, then create a new node with new key value and insert it into the tree as a root node.
* **Step 3 -**If tree is **Not Empty**, then find the suitable leaf node to which the new key value is added using Binary Search Tree logic.
* **Step 4 -**If that leaf node has empty position, add the new key value to that leaf node in ascending order of key value within the node.
* **Step 5 -**If that leaf node is already full, **split** that leaf node by sending middle value to its parent node. Repeat the same until the sending value is fixed into a node.
* **Step 6 -**If the spilting is performed at root node then the middle value becomes new root node for the tree and the height of the tree is increased by one.

##### Example

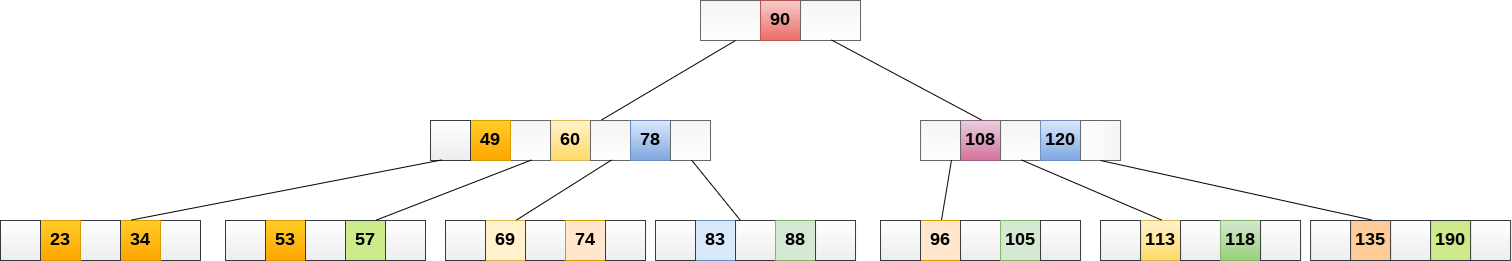
Construct a **B-Tree of Order 3** by inserting numbers from 1 to 10.



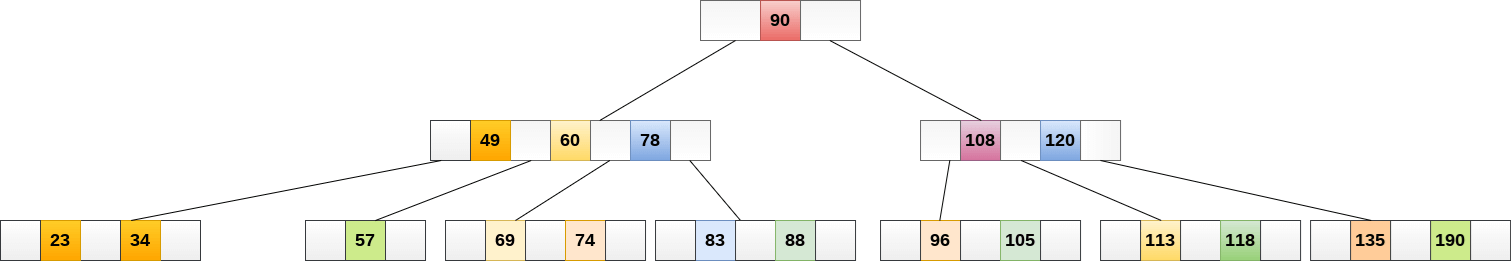


**DELETION of Elements from B Tree**

Delete the node 53 from the B Tree of order 5 shown in the following figure.



53 is present in the right child of element 49. Delete it.



Now, 57 is the only element which is left in the node, the minimum number of elements that must be present in a B tree of order 5, is 2. it is less than that, the elements in its left and right sub-tree are also not sufficient therefore, merge it with the left sibling and intervening element of parent i.e. 49.

The final B tree after deleting 53 is shown as follows.

