Problem Definition: Implement Dynamic Order Statistics using Red Black Tree.

→We want to support the usual dynamic-set operations from R-B trees, plus:

- OS-SELECT*(x, i)*: return pointer to node containing the *i* th smallest key of the subtree rooted at *x*.
- OS-RANK*(T, x)*: return the rank of *x* in the linear order determined by an inorder walk of *T* .

Technology Stack : Problem Solved In Net-Beans IDE V8.2,Using Java JDK 8.

Classes And Functions:

Classes-

1. RBtree
2. RB
3. Node

Functions In each Class-

1. **RBtree- public static void main :**This Is main Class and contains static main and works as driver for the whole program.
2. **RB -**
    a. **Insert:** It inserts the node according to BST logic and assigns it red colourand then calls function insert_check to validate the tree and colour of nodes.
    b. **Insert_check:**It checks all the cases for rbtree insertion and changes the colour of nodes accordingly.
    c. **Delete & Delete_check:** It deletes the node from rbtree validate the tree structure after deleting the node such as checking colors of the node,etc.
    d. **select:** return pointer to node containing the ith smallest key of the subtree rooted at x.
    e. **rank:**return the rank of x in linear order determined by an inorder walk of tree T.
    f. **inorder:**it is called when inorder traversal of tree is required.
    g. **Print & print_node :**to display the tree structure.
3. **Node-**
    a. Node(constructor): It assign values to every object created.

Key Functionality: This project aims on order statistics on Red Black Tree .