```java
/* PRACTICAL-6: Implementation of Binary Search Trees.

Name: Angat Shah
Enrollment No: 202203103510097
Branch: B.Tech Computer Science and Engineering */

// CODE:

import java.util.Scanner;
class TreeNode {
    int data;
    TreeNode left;
    TreeNode right;
    public TreeNode(int data) {
        this.data = data;
        this.left = null;
        this.right = null;
    }
}
class BinarySearchTree {
    private TreeNode root;
    public BinarySearchTree() {
        root = null;
    }
    public void insert(int data) {
        root = insertRec(root, data);
    }
    private TreeNode insertRec(TreeNode root, int data) {
        if (root == null) {
            root = new TreeNode(data);
            return root;
        }
        if (data < root.data)
            root.left = insertRec(root.left, data);
        else if (data > root.data)
            root.right = insertRec(root.right, data);

        return root;
    }
    public void inorder() {
        inorderRec(root);
    }
    private void inorderRec(TreeNode root) {
        if (root != null) {
            inorderRec(root.left);
            System.out.print(root.data + " ");
            inorderRec(root.right);
        }
    }
    public void preorder() {
        preorderRec(root);
    }
    private void preorderRec(TreeNode root) {
        if (root != null) {
            System.out.print(root.data + " ");
            preorderRec(root.left);
            preorderRec(root.right);
        }
```

```
    }
    public void postorder() {
        postorderRec(root);
    }
    private void postorderRec(TreeNode root) {
        if (root != null) {
            postorderRec(root.left);
            postorderRec(root.right);
            System.out.print(root.data + " ");
        }
    }
}
public class practical6 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        BinarySearchTree bst = new BinarySearchTree();
        int choice;
        do {
            System.out.println("\nBinary Search Tree Operations:");
            System.out.println("1. Insert");
            System.out.println("2. Inorder Traversal");
            System.out.println("3. Preorder Traversal");
            System.out.println("4. Postorder Traversal");
            System.out.println("5. Exit");
            System.out.print("Enter your choice: ");
            choice = scanner.nextInt();
            switch (choice) {
                case 1:
                    System.out.print("Enter element to insert: ");
                    int element = scanner.nextInt();
                    bst.insert(element);
                    break;
                case 2:
                    System.out.print("Inorder Traversal: ");
                    bst.inorder();
                    System.out.println();
                    break;
                case 3:
                    System.out.print("Preorder Traversal: ");
                    bst.preorder();
                    System.out.println();
                    break;
                case 4:
                    System.out.print("Postorder Traversal: ");
                    bst.postorder();
                    System.out.println();
                    break;
                case 5:
                    System.out.println("Exiting...");
                    break;
                default:
                    System.out.println("Invalid choice. Please enter a valid option.");
            }
        } while (choice != 5);
        scanner.close();
    }
}
```