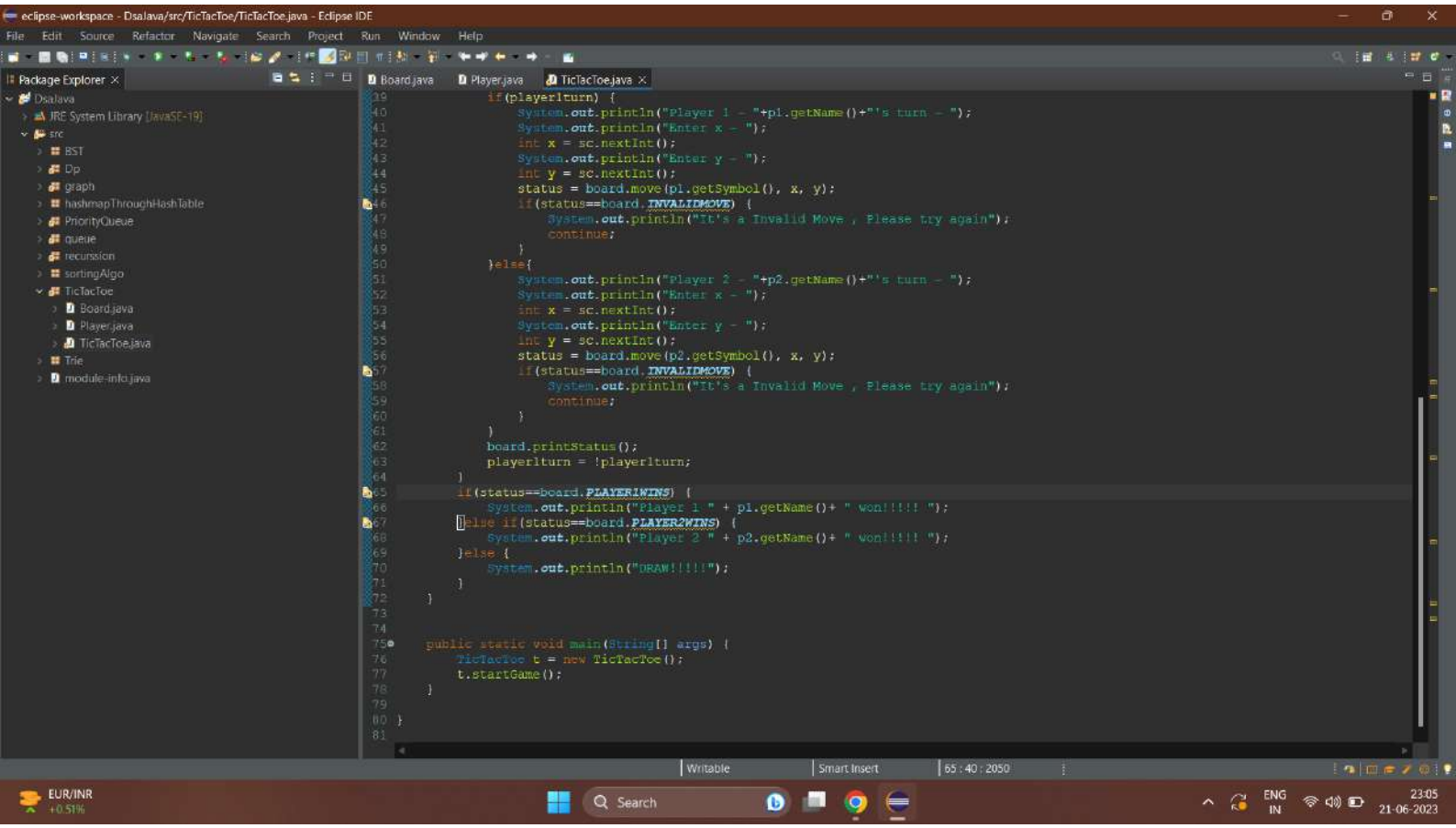
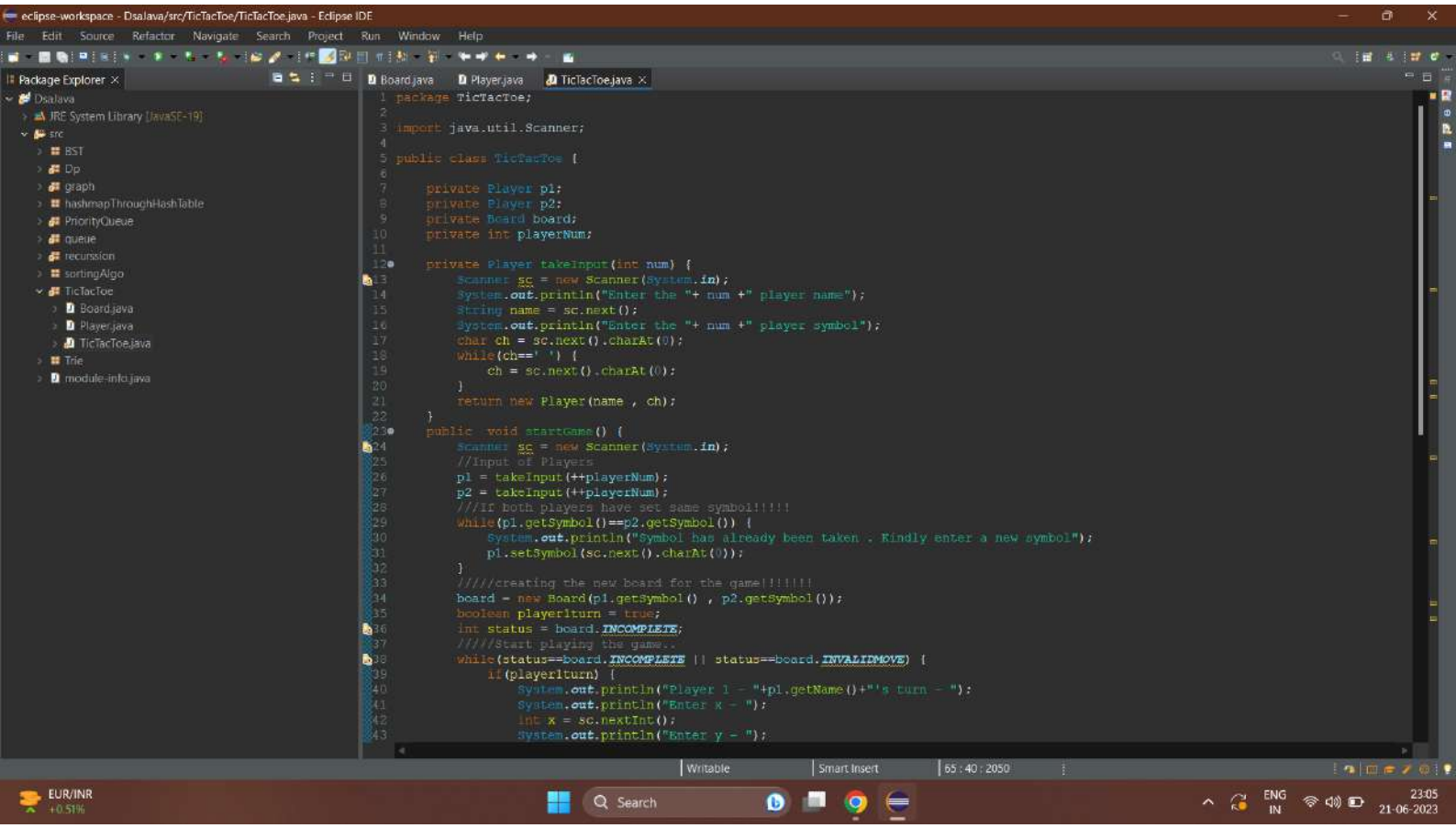
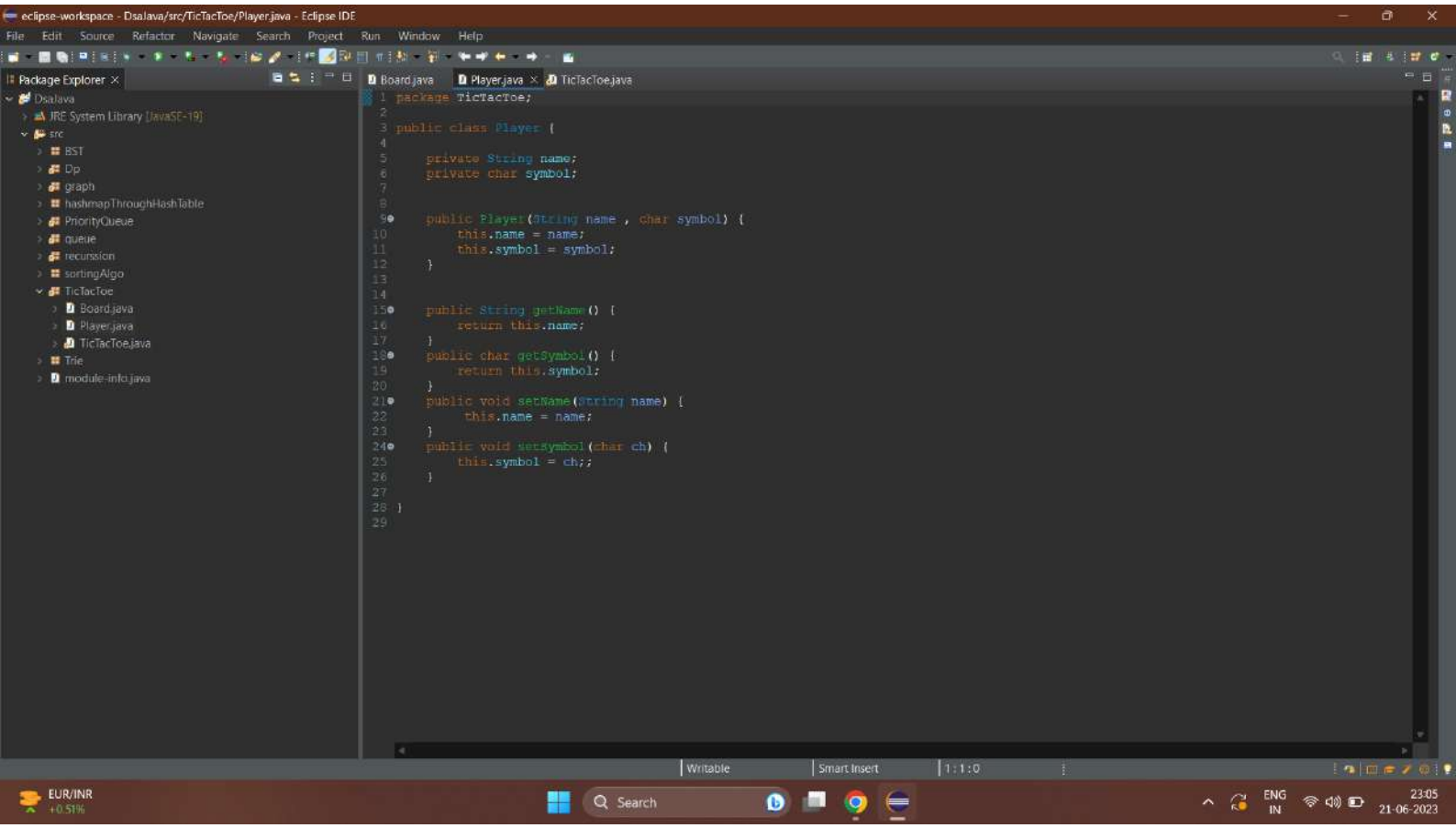
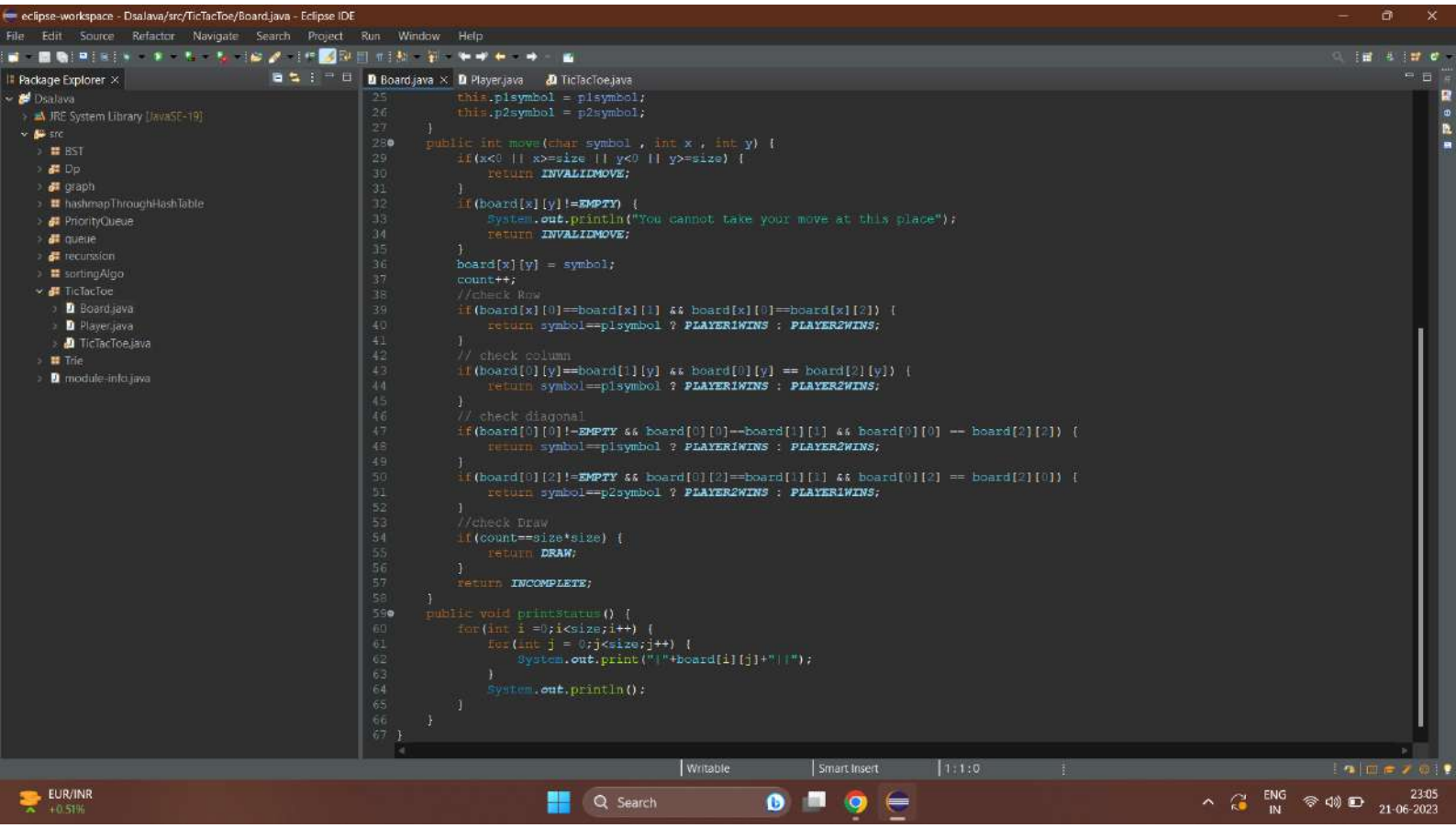
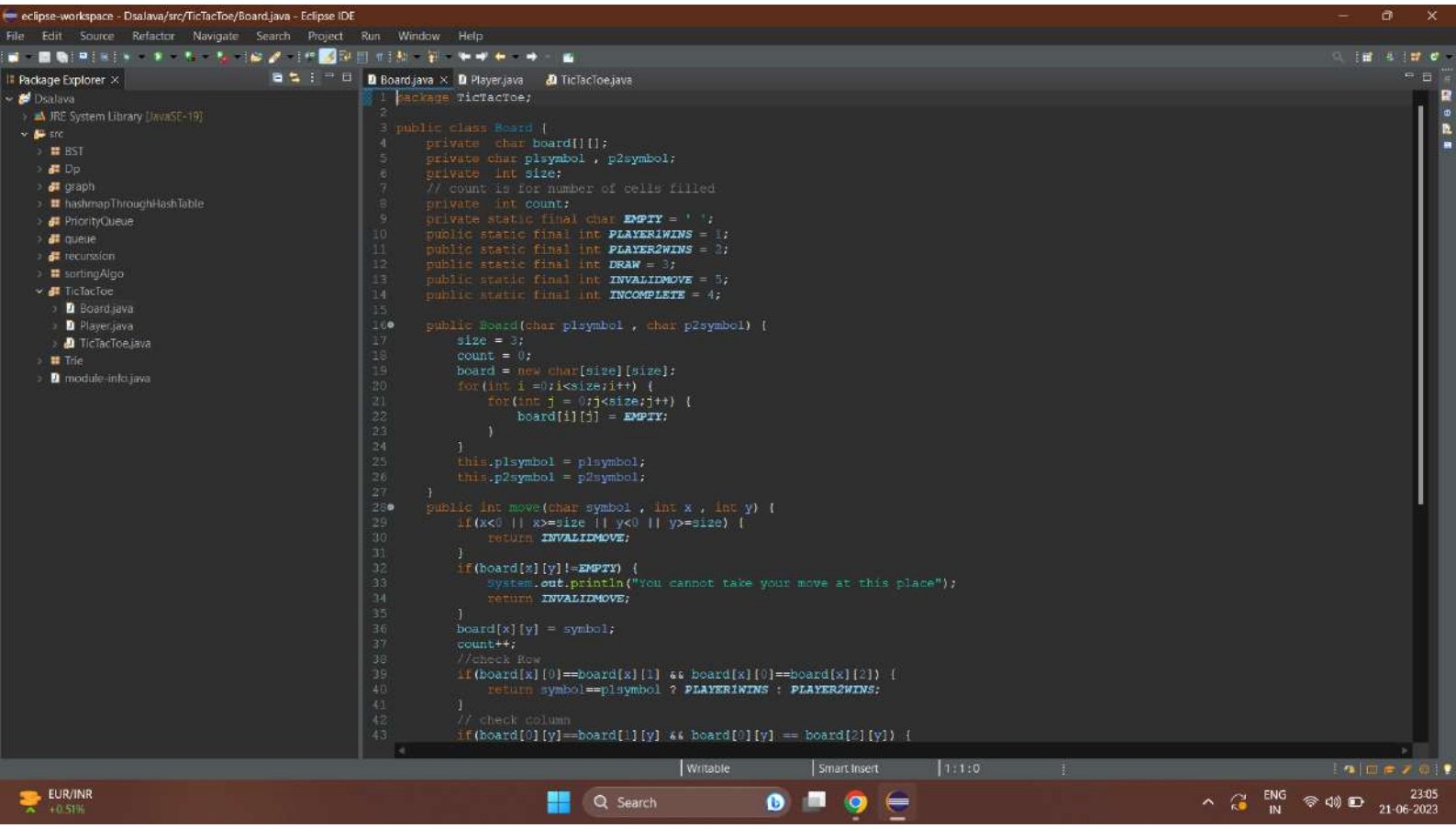

eclipse-workspace - Dsalava/src/TicTacToe/TicTacToe.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Dsalava
JRE System Library [JavaSE-19]
src
BST
Dp
graph
hashmapThroughHashTable
PriorityQueue
queue
recursion
sortingAlgo
TicTacToe
Board.java
Player.java
TicTacToe.java
Trie
module-info.java
Board.java
Player.java
TicTacToe.java
if(playerIturn) {
System.out.println("Player 1 - "+p1.getName()+"'s turn - ");
System.out.println("Enter x - ");
int x = sc.nextInt();
System.out.println("Enter y - ");
int y = sc.nextInt();
status = board.move(p1.getSymbol(), x, y);
if(status==board.INVALIDMOVE) {
System.out.println("It's a Invalid Move , Please try again");
continue;
}
}else{
System.out.println("Player 2 - "+p2.getName()+"'s turn - ");
System.out.println("Enter x - ");
int x = sc.nextInt();
System.out.println("Enter y - ");
int y = sc.nextInt();
status = board.move(p2.getSymbol(), x, y);
if(status==board.INVALIDMOVE) {
System.out.println("It's a Invalid Move , Please try again");
continue;
}
}
board.printStatus();
playerIturn = !playerIturn;
}
if(status==board.PLAYER1WINS) {
System.out.println("Player 1 " + p1.getName()+ " won!!!!! ");
}else if(status==board.PLAYER2WINS) {
System.out.println("Player 2 " + p2.getName()+ " won!!!!! ");
}else {
System.out.println("DRAW!!!!!");
}
}
public static void main(String[] args) {
TicTacToe t = new TicTacToe();
t.startGame();
}
}
Writable
Smart Insert
65 : 40 : 2050
EUR/INR
+0.51%
Search
ENG
IN
23:05
21-06-2023


eclipse-workspace - Dsalava/src/TicTacToe/TicTacToe.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Dsalava
JRE System Library [JavaSE-19]
src
BST
Dp
graph
hashmapThroughHashTable
PriorityQueue
queue
recursion
sortingAlgo
TicTacToe
Board.java
Player.java
TicTacToe.java
Trie
module-info.java
Board.java
Player.java
TicTacToe.java
1 package TicTacToe;
2
3 import java.util.Scanner;
4
5 public class TicTacToe {
6
7 private Player p1;
8 private Player p2;
9 private Board board;
10 private int playerNum;
11
12 private Player takeInput(int num) {
13 Scanner sc = new Scanner(System.in);
14 System.out.println("Enter the "+ num +" player name");
15 String name = sc.next();
16 System.out.println("Enter the "+ num +" player symbol");
17 char ch = sc.next().charAt(0);
18 while(ch==' ') {
19 ch = sc.next().charAt(0);
20 }
21 return new Player(name , ch);
22 }
23 public void startGame() {
24 Scanner sc = new Scanner(System.in);
25 //Input of Players
26 p1 = takeInput(++playerNum);
27 p2 = takeInput(++playerNum);
28 ///If both players have set same symbol!!!!!
29 while(p1.getSymbol()==p2.getSymbol()) {
30 System.out.println("Symbol has already been taken . Kindly enter a new symbol");
31 p1.setSymbol(sc.next().charAt(0));
32 }
33 /////creating the new board for the game!!!!!!
34 board = new Board(p1.getSymbol() , p2.getSymbol());
35 boolean player1turn = true;
36 int status = board.INCOMPLETE;
37 /////Start playing the game..
38 while(status==board.INCOMPLETE || status==board.INVALIDMOVE) {
39 if(player1turn) {
40 System.out.println("Player 1 - "+p1.getName()+"'s turn - ");
41 System.out.println("Enter x - ");
42 int x = sc.nextInt();
43 System.out.println("Enter y - ");
Writable
Smart Insert
65 : 40 : 2050
EUR/INR
+0.51%
Search
ENG
IN
23:05
21-06-2023


eclipse-workspace - Dsalava/src/TicTacToe/Player.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Dsalava
JRE System Library [JavaSE-19]
src
BST
Dp
graph
hashmapThroughHashTable
PriorityQueue
queue
recursion
sortingAlgo
TicTacToe
Board.java
Player.java
TicTacToe.java
Trie
module-info.java
Board.java
Player.java
TicTacToe.java
1 package TicTacToe;
2
3 public class Player {
4
5 private String name;
6 private char symbol;
7
8
9 public Player(String name , char symbol) {
10 this.name = name;
11 this.symbol = symbol;
12 }
13
14
15 public String getName() {
16 return this.name;
17 }
18 public char getSymbol() {
19 return this.symbol;
20 }
21 public void setName(String name) {
22 this.name = name;
23 }
24 public void setSymbol(char ch) {
25 this.symbol = ch;;
26 }
27
28 }
29
Writable
Smart Insert
1 : 1 : 0
EUR/INR
+0.51%
Search
ENG
IN
23:05
21-06-2023


eclipse-workspace - Dsalava/src/TicTacToe/Board.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Dsalava
JRE System Library [JavaSE-19]
src
BST
Dp
graph
hashmapThroughHashTable
PriorityQueue
queue
recursion
sortingAlgo
TicTacToe
Board.java
Player.java
TicTacToe.java
Trie
module-info.java
Board.java
Player.java
TicTacToe.java
this.p1symbol = p1symbol;
this.p2symbol = p2symbol;
}
public int move(char symbol , int x , int y) {
if(x<0 || x>=size || y<0 || y>=size) {
return INVALIDMOVE;
}
if(board[x][y]!=EMPTY) {
System.out.println("You cannot take your move at this place");
return INVALIDMOVE;
}
board[x][y] = symbol;
count++;
//check Row
if(board[x][0]==board[x][1] && board[x][0]==board[x][2]) {
return symbol==p1symbol ? PLAYER1WINS : PLAYER2WINS;
}
// check column
if(board[0][y]==board[1][y] && board[0][y] == board[2][y]) {
return symbol==p1symbol ? PLAYER1WINS : PLAYER2WINS;
}
// check diagonal
if(board[0][0]!=EMPTY && board[0][0]==board[1][1] && board[0][0] == board[2][2]) {
return symbol==p1symbol ? PLAYER1WINS : PLAYER2WINS;
}
if(board[0][2]!=EMPTY && board[0][2]==board[1][1] && board[0][2] == board[2][0]) {
return symbol==p2symbol ? PLAYER2WINS : PLAYER1WINS;
}
//check Draw
if(count==size*size) {
return DRAW;
}
return INCOMPLETE;
}
public void printStatus() {
for(int i =0;i<size;i++) {
for(int j = 0;j<size;j++) {
System.out.print("|"+board[i][j]+"|");
}
System.out.println();
}
}
}
Writable
Smart Insert
1 : 1 : 0
EUR/INR
+0.51%
Search
ENG
IN
23:05
21-06-2023


eclipse-workspace - Dsalava/src/TicTacToe/Board.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Dsalava
JRE System Library [JavaSE-19]
src
BST
Dp
graph
hashmapThroughHashTable
PriorityQueue
queue
recursion
sortingAlgo
TicTacToe
Board.java
Player.java
TicTacToe.java
Trie
module-info.java
Board.java
Player.java
TicTacToe.java
1 package TicTacToe;
2
3 public class Board {
4 private char board[][];
5 private char p1symbol , p2symbol;
6 private int size;
7 // count is for number of cells filled
8 private int count;
9 private static final char EMPTY = ' ';
10 public static final int PLAYER1WINS = 1;
11 public static final int PLAYER2WINS = 2;
12 public static final int DRAW = 3;
13 public static final int INVALIDMOVE = 5;
14 public static final int INCOMPLETE = 4;
15
16 public Board(char p1symbol , char p2symbol) {
17 size = 3;
18 count = 0;
19 board = new char[size][size];
20 for(int i =0;i<size;i++) {
21 for(int j = 0;j<size;j++) {
22 board[i][j] = EMPTY;
23 }
24 }
25 this.p1symbol = p1symbol;
26 this.p2symbol = p2symbol;
27 }
28 public int move(char symbol , int x , int y) {
29 if(x<0 || x>=size || y<0 || y>=size) {
30 return INVALIDMOVE;
31 }
32 if(board[x][y]!=EMPTY) {
33 System.out.println("You cannot take your move at this place");
34 return INVALIDMOVE;
35 }
36 board[x][y] = symbol;
37 count++;
38 //check Row
39 if(board[x][0]==board[x][1] && board[x][0]==board[x][2]) {
40 return symbol==p1symbol ? PLAYER1WINS : PLAYER2WINS;
41 }
42 // check column
43 if(board[0][y]==board[1][y] && board[0][y] == board[2][y]) {
Writable
Smart Insert
1 : 1 : 0
EUR/INR
+0.51%
Search
ENG
IN
23:05
21-06-2023

Coding Ninjas | Maximum Sum BST in Binary Tree | Problems - LeetCode

leetcode.com/problems/maximum-sum-bst-in-binary-tree/description/

Gmail | Coding Ninjas | Online Java Compil... | FSA: Inbox | Problems - LeetCode | FINAL450.xlsx - Go... | The Only Lists You... | Find Common Char... | Merge Intervals - L... | K-th Smallest Prime...

LeetCode | Problem List | Premium

Description | Editorial | Solutions (749) | Submissions

Constraints:

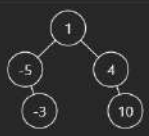
- The number of nodes in the tree is in the range `[1, 4 * 10^4]`.
- `-4 * 10^4 <= Node.val <= 4 * 10^4`

Testcase | Result | Source

Case 1 | Case 2

root =

```
[1,-5,4,null,-3,null,10]
```



Console | Reset Testcases | Run | Submit

Java | Auto

```java
class Solution {
    public pair fun(TreeNode root){
        if(root==null){
            return new pair(true , Integer.MAX_VALUE , Integer.MIN_VALUE , 0);
        }
        pair left = fun(root.left);
        pair right = fun(root.right);
        boolean isBst = true;
        int sum = 0;
        if(root.val<=left.max){
            isBst = false;
        }
        if(root.val>=right.min){
            isBst = false;
        }
        if(left.isBst && right.isBst && (isBst)){
            int total = left.sum + right.sum + root.val;
            sum = Math.max(left.sum , right.sum);
            sum = Math.max(sum , total);
        }else{
            isBst = false;
            sum = Math.max(left.sum , right.sum);
        }
        int min = Math.min(root.val , Math.min(left.min , right.min));
        int max = Math.max(root.val , Math.max(left.max , right.max));
        return new pair(isBst , min , max , sum);
    }
    public int maxSumBST(TreeNode root) {
        pair result = fun(root);
        int sum =  result.sum;
        if(sum<=0){
            return 0;
        }
        return sum;
```

19°C Mostly cloudy | Search | ENG IN | 05:03 26-03-2023