**package** balanced\_brackets.java;

**import** java.util.\*;

**import** java.util.Scanner;

**public** **class** BalancedBrackets {

// function to check whether brackets are balanced

**static** **boolean** areBracketsBalanced(String str)

{

// Using ArrayDeque method

Deque<Character> stack= **new** ArrayDeque<Character>();

// Traverse Expression

**for** (**int** i = 0; i < str.length(); i++)

{

**char** x = str.charAt(i);

**if** (x == '(' || x == '[' || x == '{')

{

// Pushing element into stack

stack.push(x);

**continue**;

}

// If current character is not the opening

// bracket, then it may be closing. So stack

// cannot be empty

**if** (stack.isEmpty())

**return** **false**;

**char** check;

**switch** (x) {

**case** ')':

check = stack.pop();

**if** (check == '{' || check == '[')

**return** **false**;

**break**;

**case** '}':

check = stack.pop();

**if** (check == '(' || check == '[')

**return** **false**;

**break**;

**case** ']':

check = stack.pop();

**if** (check == '(' || check == '{')

**return** **false**;

**break**;

}

}

// Check Empty Stack

**return** (stack.isEmpty());

}

// Driver code

**public** **static** **void** main(String[] args)

{

System.***out***.println("Enter brackets to be checked for balance "+ "");

Scanner sc = **new** Scanner(System.***in***);

String str = sc.nextLine();

// final function calling if/else

**if** (*areBracketsBalanced*(str))

System.***out***.println("The entered String has Balanced Brackets ");

**else**

System.***out***.println("The entered Strings do not contain Balanced Brackets ");

**return**;

}

}