

```

package stacksqueues;

// Java program for checking
// balanced brackets
import java.util.*;

public class _1BalancedBrackets {

    // function to check if brackets are balanced
    static boolean areBracketsBalanced(String expr)
    {
        // Using ArrayDeque is faster than using Stack class
        Deque<Character> stack
            = new ArrayDeque<Character>();

        // Traversing the Expression
        for (int i = 0; i < expr.length(); i++)
        {
            char x = expr.charAt(i);

            if (x == '(' || x == '[' || x == '{')
            {
                // Push the element in the stack
                stack.push(x);
                continue;
            }

            // IF current current character is not opening
            // bracket, then it must be closing. So stack
            // cannot be empty at this point.
            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)
        {
            String expr = "([{}])";

            // Function call
            if (areBracketsBalanced(expr))
                System.out.println("Balanced ");
        }
    }
}

```

```
        else
            System.out.println("Not Balanced ");
    }
}
```